

一种服务 Agent 的可信性评估方法*

朱曼玲^{1,3}, 金芝^{1,2+}

¹(中国科学院 数学与系统科学研究院, 北京 100190)

²(高可信软件技术教育部重点实验室(北京大学), 北京 100871)

³(对外经济贸易大学 网络与教育技术中心, 北京 100029)

Approach for Evaluating the Trustworthiness of Service Agent

ZHU Man-Ling^{1,3}, JIN Zhi^{1,2+}

¹(Academy of Mathematics and Systems Science, The Chinese Academy of Sciences, Beijing 100190, China)

²(Key Laboratory of High Confidence Software Technologies (Peking University), Beijing 100871, China)

³(Network and Educational Technology Center, University of International Business and Economics, Beijing 100029, China)

+ Corresponding author: E-mail: zhijin@amss.ac.cn

Zhu ML, Jin Z. Approach for evaluating the trustworthiness of service agent. *Journal of Software*, 2011, 22(11):2593-2609. <http://www.jos.org.cn/1000-9825/3921.htm>

Abstract: This paper addresses the service selection issue via a service agent framework coupled with a computational model based on trust ontology. The trust ontology proposed from the socio-cognitive view can offer a more insightful understanding in the trust relationships among service agents. A set of trust computational rules have been given to support the trust analysis and reasoning. Also the structure of the service agents has been given. Finally, a case study is used to illustrate how the service agents infer the trustworthiness of their partners by using the set of trust reasoning and computational rules.

Key words: service-oriented computing; service agent; trust ontology; socio-cognitive approach

摘要: 提出了一个基于服务 Agent 的计算框架,并从社交认知的角度建立了一个服务 Agent 的信任本体,支持服务 Agent 对信任信息进行推理.根据该信任本体,提出一系列基于信任推理的计算规则支持信任值的计算,帮助服务 Agent 进行理性的选择决策.案例研究结果表明,该方法能够有效地帮助服务请求方进行信任评价以及服务选择.

关键词: 面向服务计算;服务 Agent;信任本体;社交认知方法

中图法分类号: TP311 文献标识码: A

由于 Web 服务的广泛应用,越来越多的专业服务提供商在服务注册中心发布自己的服务,使得服务请求方有可能在服务注册中心发现多个功能描述相似的服务信息能够与其需求匹配;另一方面,服务提供方则有可能由于自身利益的驱动,发布虚假或者不充分的服务信息,使得基于功能匹配的服务选择绑定彻底失败.于是,服务请求方就必然希望选择可信的服务,完成服务请求与绑定.因此,随着 Internet 的普及和面向服务计算发展规模的不断扩大,服务选择和提供中的可信性问题变得越来越重要.

* 基金项目: 国家自然科学基金 (90818026); 国家重点基础研究发展计划(973)(2009CB320701); 国家杰出青年基金(60625204)

收稿时间: 2010-01-31; 修改时间: 2010-05-05; 定稿时间: 2010-07-30

传统的面向服务计算领域对服务可信性的研究主要关注于服务的质量(QoS)^[1],认为服务质量的高低直接影响服务的可信性,即服务质量参数决定可信与否,质量好的服务更为可信^[2,3].另一方面,结合 Agent 的研究,Maximilien 等人用服务质量作为信任评估的依据,将实际的服务在其质量各个维度上达到服务提供方所宣称的程度,认为是服务可信性的程度,提出用 Agent 帮助服务提供方与服务请求方进行动态服务选择^[4].实际上,在 Internet 环境下的服务选择过程从两个方面体现了 Web 服务本身所具有的自主性:首先,发起服务请求后,对多个功能相似的服务的选择显示出服务请求的自主性;其次,服务提供方是否愿意被服务请求调用并与其绑定,显示了服务提供方同样具有其自主性.现有 SOA 解决方案在刻画服务请求和服务提供的自主方面有很大不足,而 Web 服务的自主性会对基于服务信任关系的选择与绑定产生影响.

已经有一些工作将 Agent 技术引入面向服务计算领域.Hendler^[5]在 2001 年就提出 Agent 技术的引入会给 Web 服务带来深远的影响;在 Huhns 等人的工作^[6,7]中,服务被看作是能够以自治灵活的行为方式达成其自身目标的 Agent;Sycara 等研究人员^[8]也提出在开放网络环境下多 Agent 协作机制可以帮助 Web 服务进行服务匹配和绑定.服务由传统的被动实体转变为主动实体,既包含 Web 服务本身(服务实现体),也具备 Agent 的自主性等各种特性,具有目标、信念和一些要满足的需求、核心能力以及满足自身需求的相关知识^[9],我们称其为服务 Agent.面向服务的系统可以看作是由这样的服务 Agents 构成.服务选择和提供中的可信性问题也就是服务 Agents 的可信性问题.

在 Agent 信任建模领域,对个体 Agent 进行可信性评估的工作包括:

- (1) 演化和学习模型.发起请求的 Agent 在持续的交互过程中,基于自己有限的经验对信任进行度量,选择交互对象,学习对方的策略,应对说谎或恶意的 Agent,以追求更多的长期回报.如 Mui 等人的信任模型^[10].
- (2) 声誉模型.建立收集其他 Agent 信息的手段,并能够从已取得的被评估 Agent 的信息中进行合理地推理.声誉模型的核心是收集信息与对证人撒谎的识别与处理,如 Sabater 和 Sierra 的模型^[11].
- (3) 社交认知模型.建立评估对象的思维状态模型,从动机、能力等方面确定是否相信对方会诚实地进行交互,如 Castelfranchi 和 Falcone 的模型^[12].

服务 Agent 具有自主、自利和社会的特性,能够根据已有知识进行推理和判断.因此,对服务 Agent 的思维状态及行为表现的刻画与分析相当重要,但单从 Agent 信任建模角度出发对服务 Agent 进行信任评估是不能满足要求的,这是因为服务 Agent 是分布在网络上的服务应用,具有特定的功能,也就是服务 Agent 只可能具有有限能力,而在多 Agent 领域中缺乏对 Agent 能力的考察和分析,认为 Agent 根据自己的目标可以完成任何任务.因此,纯 Agent 领域可信性建模方法不能直接用于服务选择中的信任分析.建立服务有限能力与 Agent 信任建模方法相结合的技术,对服务 Agent 信任评估与决策具有重要的意义.

本文提出了一个服务 Agent 信任评估框架,并给出服务 Agent 的结构及表示.从社交认知的角度为服务 Agents 之间的信任关系进行建模,抽取并规范信任评估概念体系,建立了一个信任本体,该信任本体能够支持信任的分析和推理.服务 Agents 根据信任推理结果和一组信任计算规则得到最终信任值,该信任值可以帮助服务 Agents 做出合理的服务选择决策.

本文第 1 节给出服务 Agent 的信任评估框架.第 2 节从社会认知角度建立一个信任本体.第 3 节讨论信任相关证据的表示和获取方式.第 4 节定义一组信任计算规则,并进行数值模拟,最后结合具体案例讨论工具实现.第 5 节给出与相关工作的比较.第 6 节总结全文并提出进一步的工作.

1 服务 Agent 信任评估框架

1.1 基于服务 Agent 的服务请求交互框架

服务 Agent 是分布于 Internet 环境下各个节点上、具有主体化特征的软件实体.服务 Agent 具有一定的服务功能,可以完成特定的任务.当其需要其他服务时也可以发出服务请求,并根据自身感知、分析和推理能力做出选择决策.基于服务 Agent 的服务请求交互过程如下:一个服务 Agent 首先根据自己的目标发起服务请求,一

些服务 Agents 对该服务请求做出响应.于是,服务请求方通过从外部环境感知到的证据信息以及自身已有知识对这些响应请求的服务 Agents 进行评估,比较评估结果,进行服务选择,告知所有响应服务请求的服务 Agents,最终选中的服务 Agent 为该请求方提供其所需服务.图 1 给出了基于服务 Agent 的服务请求交互框架.4 种箭头代表了服务 Agents 之间 4 种服务请求交互类型:请求方发出服务请求、提供方响应服务请求、请求方告知服务选择结果(接受或拒绝)以及提供方提供服务.

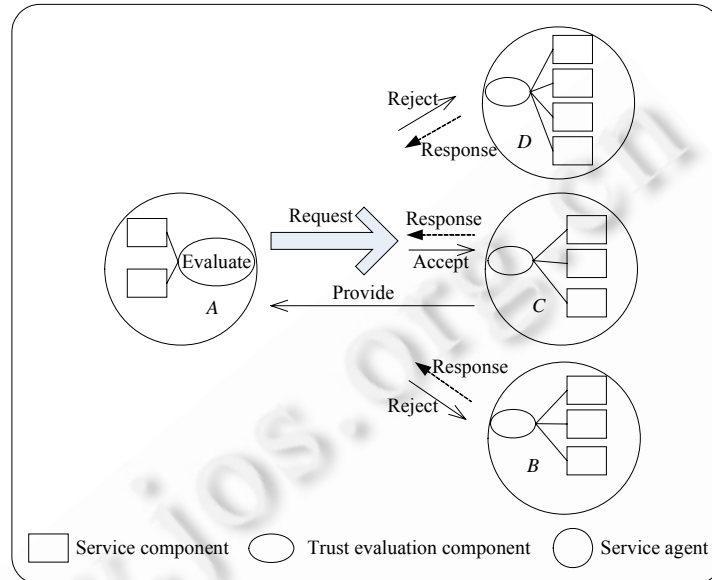


Fig.1 Framework of service agent based interaction

图 1 基于服务 Agent 的服务请求交互框架

1.2 服务 Agent 结构及其形式化表示

图 2 给出了服务 Agent 的结构,它由两部分组成:一部分是服务构件,另一部分是信任评估构件.

服务构件提供服务的能力,包含服务实现部分和服务描述.服务实现部分即服务软件模块.它部署在由服务提供者提供的可以通过网络访问的平台上,服务软件模块的相关内容不在本文的讨论范围内.服务描述包含目标集、可用资源集、知识技能集和该服务 Agent 的声誉值,可以通过网络被其他服务 Agent 感知.服务描述各部分的具体含义将在第 2 节中作详细说明.

信任评估构件包含 3 个部分:知识和事实库、推理引擎和感知/动作接口.其中,知识和事实库又包含信任本体库、证据信息库和计算策略库,这部分内容将在第 2 节~第 4 节分别给出详细定义和解释.推理引擎包括信息更新模块、信任信念推理模块、信念值更新模块、信任值产生模块和可信性判断模块.服务 Agent 从外部感知到的关于被评估服务 Agent 的信息及证据信息库中的已有信息共同作为信息更新模块的输入,输出更新后的综合信息.这些信息可返回证据信息库以更新其中的记录,也可与信任本体库中的推理规则一起作为信任信念推理模块的输入.信任信念推理模块用于进行信任信念推理,输出的推理结果与证据信息库中的已有信念值及服务 Agent 计算策略库中的信任信念计算规则一起作为信念值更新模块的输入.服务 Agent 通过该模块进行信任信念计算,得到信念值更新结果,更新后的信念值、计算策略库中的信任信念偏好和信任值综合算法作为信任值产生模块的输入,得到被评估对象的最终信任评估结果.可信性判断模块用于将所有被评估对象的信任值加以比较,得出理性的选择决策.感知/动作接口包括感知模块和行为模块,服务 Agent 通过感知模块感知外部事件,通过行为模块输出服务 Agent 的理性行为.

服务 Agent 可以承担两种不同的角色:当其根据自己的需要发起服务请求时作为服务请求方,当其响应其

他服务 Agent 的服务请求时又可以作为服务提供方.为了便于在信任分析过程中针对其角色特点进行研究,我们分角色给出服务 Agent 的定义.

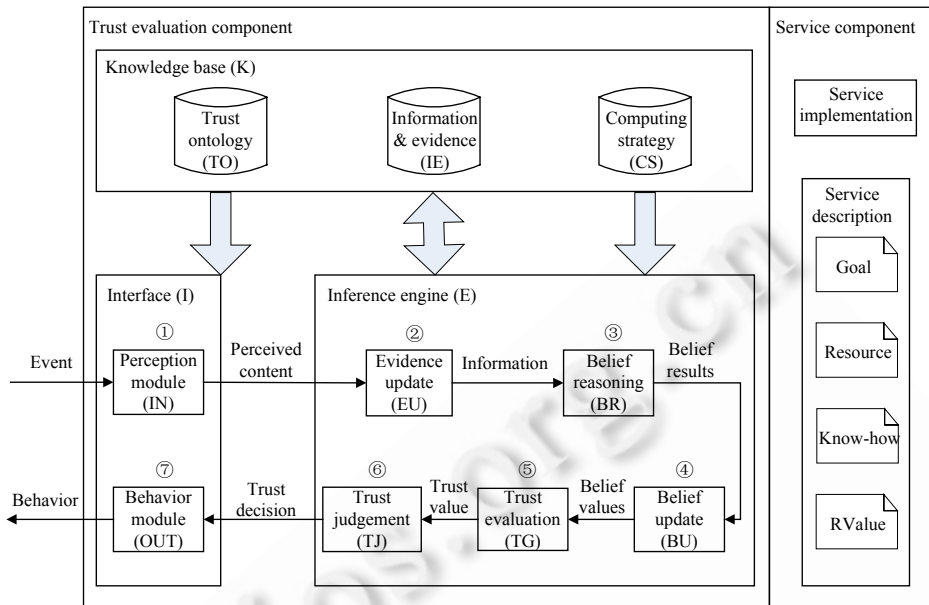


Fig.2 Structure of a service agent

图2 服务 Agent 结构

定义 1. 一个服务请求 Agent 可以表示为一个三元组:

$$\text{服务请求 Agent} = \langle K, E, I \rangle,$$

其中, $K = \langle TO, IE, CS \rangle$ 是服务请求 Agent 的知识和事实库, 包含信任本体库、证据信息库和计算策略库 3 个部分; $E = \langle EU, BR, BU, TG, TJ \rangle$ 是服务请求 Agent 用于信任分析评估过程的推理引擎; $I = \langle IN, OUT \rangle$ 是服务请求 Agent 的感知/动作接口, 包含感知模块和行为模块两部分.

定义 2. 一个服务提供 Agent 可以表示为一个四元组:

$$\text{服务提供 Agent} = \langle Goal, Resource, Know-how, RValue \rangle,$$

其中, $Goal$ 是服务提供 Agent 具有的需要达成的目标集合, $Resource$ 是服务提供 Agent 具有的资源集合, $Know-how$ 是服务提供 Agent 具有的知识技能集合, $RValue$ 是服务提供 Agent 具有的公众声誉评价价值.

1.3 服务 Agent 信任评估框架和过程

服务 Agents 进行选择时, 请求方需要考虑服务提供方的信任程度以保证得到满意、可信的服务. 作为请求方的服务 Agent 需要对可能的提供方进行客观的信任分析、推理和评估, 最后根据自己的主观偏好选出满意的服务提供方.

为了使服务 Agents 对信任关系建立过程中需要的相关知识达成统一的理解和认识, 我们提出一个信任本体, 该信任本体包含信任维度(即信念概念)和信任推理相关概念、概念间关联关系以及一组信任推理规则. 服务 Agents 通过直接交互方式、观察方式和基于大样本源的询问方式获取相关证据信息. 服务 Agents 根据信任本体中的信任维度推理规则和证据信息分析信任信念, 最后通过一组信任信念计算策略及其自身的主观偏好选择可信的服务提供方. 图 3 是基于信任本体的信任评估与决策模型. 信任本体给出了进行信任分析推理过程的知识基础; 服务 Agents 通过多种方式获得的信任信息给出了信任分析推理的证据基础; 根据信任信念推理规则从推理信息中推演信任信念; 然后将推理判断得到的信任信念算法化, 并综合信任各维度得出信任值; 最后, 服

务 Agents 对比所有潜在交互对象的信任值作判断决策。

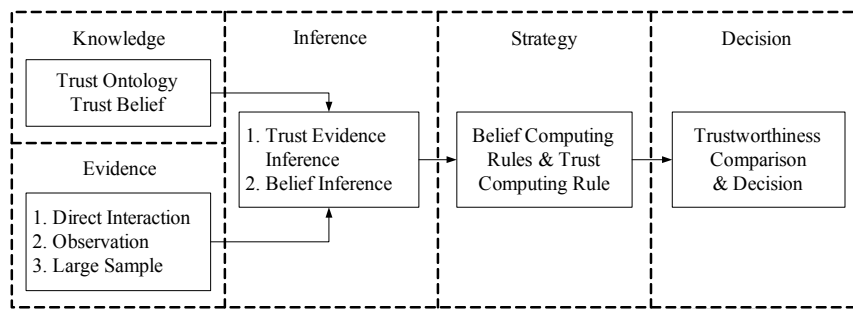


Fig.3 Evaluation and decision of trust based on a trust ontology

图3 基于信任本体的信任评估与决策

一个服务 Agent 向其他服务 Agents 发起服务请求之后,一些服务 Agents 将会响应该服务请求.该服务请求方的一次信任分析过程包含以下步骤:

- (1) 请求方需要考查这些提供方是否可信,首先通过观察对方或询问其他服务 Agents 获得被评估方的信息(图 2①).
- (2) 将所感知的信息数据与其自身信息数据库中与被评估对象以往交互证据记录相结合(图 2②).
- (3) 将综合后的所有信息依照信任本体中的推理规则进行推理,得到信任本体中定义的信任的各个维度的推理结果(图 2③).
- (4) 将推理结果和数据信息库中已有的信任维度累积数值运用信任维度计算规则加以计算(图 2④).
- (5) 根据自己的偏好及综合计算信任值的策略得到服务提供方的综合信任值(图 2⑤).
- (6) 比较各个被评估对象的信任值选择一个最可信的服务提供方(图 2⑥).
- (7) 请求方与选择出的服务提供方交互(图 2⑦).
- (8) 请求方通过此次交互,获得交互经验(图 2①),更新自己的交互知识到信息数据库(图 2②),并可向其他服务 Agents 提供信息作为未来交互的参考.

具体的交互信息和计算规则将分别在后面的章节中给出.

2 信任本体

信任是一方对另一方的一种正面的社会态度,并且这个态度会随着他们之间动态的交互状况而改变^[13,14].与人类社会中的信任态度相似,服务 Agents 之间的交互体现了其社交属性,他们之间的信任建立在服务 Agents 的社会认知和对于交互对象的信任相关信念(如能力信念、意图信念等)之上.我们采用社交认知的观点建立信任本体,用来形式化表示信任的本质含义,并使得服务 Agents 对于信任概念具有一致的理解.服务提供方基于信任本体来表达自己的信任相关信息,同时,服务请求方也根据信任本体体现自己的策略以及偏好并对信任进行评估.

我们的信任本体分为 3 个层次:上层信任本体、中层信任本体和下层信任本体.上层信任本体刻画了与信任维度(即信任信念)基本特性相关联的一些概念,以及这些概念之间的关联关系.中层信任本体具体定义了信任各个维度概念和用于推理各个维度的相关概念及其关联关系,另外还包含建立在这些概念和关联之上的一组推理规则作为中层本体的约束集;中层信任本体支持信任推理,也是信任值计算的基础.下层信任本体中的概念和关联是中层本体概念关联的实例化,为具体应用领域中的信任本体,下层信任本体将在案例研究中给出.

2.1 上层信任本体

我们在文献[15]中已经构造了一个上层信任本体,图 4 给出了上层信任本体概念及概念之间关联图.

- TrustBelief 表示从社交认知的角度来看信任的一个可计算的方面,即信任的维度.信任信念概念从社会认知的角度描述信任的本质含义,用于描述信任的维度.
- TBAttribute 描述了一个特定信任信念的类型.例如,该信任信念是一个二元决策属性类型(即该信任信念值为二元 {0,1} 值),或一个累积属性类型(即该信任信念值是通过不断累加获得的).
- TBMeasurement 表示利用直接或间接信息源取得信任信念的证据信息用于信任信念度量的方法.服务 Agents 直接度量方式基于以往自身交互经验,而间接度量方式包含从其他服务 Agents 处得到的证据信息.
- TBRelationship 描述了信任信念之间的关联关系.信任信念之间可能具有关联关系,也可能相互独立.

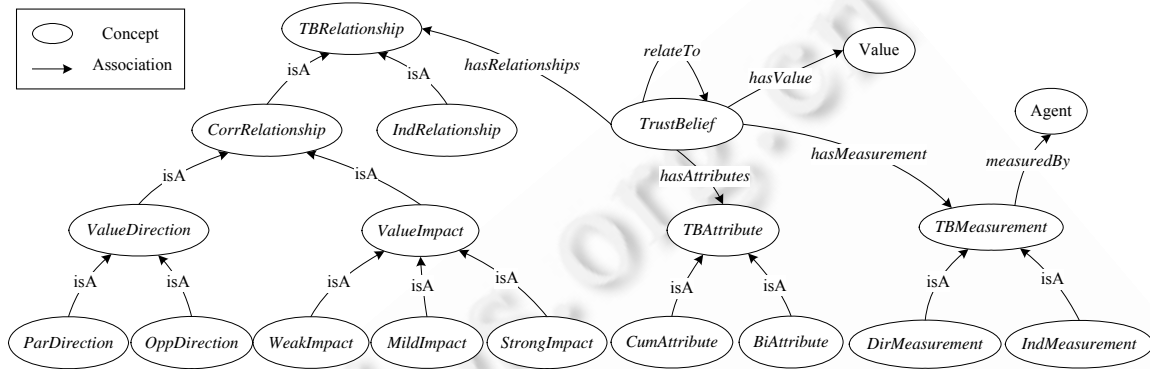


Fig.4 Trust upper ontology

图4 上层信任本体

2.2 中层信任本体

为了描述信任相关因素和用于推理信任相关因素的基本概念术语和关联,我们建立了中层信任本体.中层信任本体是信任本体的一个子本体,其概念直接关联到上层信任本体中的 **TrustBelief** 概念.中层信任本体不同于上层信任本体,中层信任本体捕获了可应用于多个领域的信任相关概念,而这些概念又可以实例化为具体领域中概念,也就是下层信任本体中的相应概念.我们给出信任子本体的定义:

定义 3(信任子本体). 信任子本体用三元组表示为

$$TrustO = (TrustC, R, A^{\circ}),$$

其中, $TrustC$ 是信任子本体概念集; R 是信任子本体的关联集合,包含用以支持信任信念推理的概念间的关联; A° 是一个公理集,包含信任本体中的概念和关联需要满足的约束,即对应我们的一组信任规则.

2.2.1 信任概念

从社交认知的观点来看,信任概念隐含着对 **Agent** 内部特征的探讨,特别是信任相关信念以及对意图和思维活动的关注.在面向服务领域中,当一个服务 **Agent** 需要另一服务 **Agent** 为它提供某一服务时,这个请求方必然要从一些方面对对方的可信性进行观察分析,然后决定是否选择其为自己提供服务.还要考虑服务 **Agent** 怎样才能得到这些方面的认识.因此,信任子本体中的概念分为两部分,分别是服务 **Agent** 需要考察提供方哪些信任有关方面,即信任信念概念,以及服务 **Agent** 依靠自身的知识和观察到的信息对对方的信任信念进行推理所涉及的信任推理概念.

从社交认知的角度,当请求方认为对方有能力并且有意图为其提供该服务时,它对提供方的信任才有可能建立.而进一步考虑,如果提供方能够遵守自己的承诺,或会对自己的意图和行为有所坚持,又或者请求方认为该提供方的行为是可预期的,那么请求方对对方的信任程度将会加深.根据以上分析,面向服务领域的信任信念概念分为 3 类:首先,一个服务请求方对于一个服务提供方具有一定的与所需服务相关的信任信念,包括对方是否有提供该服务的能力(competence),对方是否有意愿为请求方提供这项服务(intention);其次,服务请求方还有

一系列对对方性格特征上的认识,即对提供者是否会正直诚实(integrity),是否会坚持自己的目标(persistence),其行为是否可预期(predictability)具有自己的认知;最后,一个特殊的信任信念关系到请求者对服务提供者的声誉(reputation)的考察,对于不同声誉状况的交互对象,请求者可以有自己判断。

用于信任推理的基本概念包括主动自治实体服务 Agent(agent),在服务选择中它完全代理服务(service)。服务 Agent 在不同的选择场景中可以作为服务提供方(provider)或服务请求方(requester)。每个服务由一系列特定可执行行为(action)构成。每个服务 Agent 具有自己的目标(goal),目标被定义为一个事件的条件或状态。资源(resource)被定义为一个物理或信息实体,用以执行某行为或完成特定服务。知识技能(know-how)用以指导服务 Agent 的行为。完成某一可执行行为会产生一个非 0 结果(consequence),是该行为所带来的影响。

2.2.2 信任关联

信任子本体中的概念不是孤立存在的,它们之间存在联系。表 1 给出了信任概念之间的关联及其含义。

Table 1 Meanings of the associations in trust sub-ontology

表 1 信任子本体概念关联及其含义

Association	Formation	Instance	Meaning
<i>hasGoal</i>	Agent→Goal	<i>hasGoal(a,g)</i>	Agent <i>a</i> has goal <i>g</i> .
<i>hasResource</i>	Agent→Resource	<i>hasResource(a,e)</i>	Agent <i>a</i> has resource <i>e</i> .
<i>hasKnow-how</i>	Agent→Know-how	<i>hasKnow-how(a,k)</i>	Agent <i>a</i> has Know-how <i>k</i> .
<i>needResource</i>	Action→Resource	<i>needResource(act,e)</i>	Action <i>act</i> needs resource <i>e</i> .
<i>needKnow-how</i>	Action→Know-how	<i>needKnow-how(act,k)</i>	Action <i>act</i> needs know-how <i>k</i> .
<i>bringAbout</i>	Action→Consequence	<i>bringAbout(act,c)</i>	Action <i>act</i> brings about consequence <i>c</i> .
<i>achievedByService</i>	Goal→Service	<i>achievedByService(g,s)</i>	Goal <i>g</i> is achieved by service <i>s</i> .
<i>partOf</i>	Action→Service /Consequence→Goal	<i>partOf(act,s)</i> <i>/partOf(c,g)</i>	Action <i>act</i> is part of service <i>s</i> . /Consequence <i>c</i> is part of goal <i>g</i> .
<i>canAction</i>	Agent→Action	<i>canAction(a,act)</i>	Agent <i>a</i> can do action <i>act</i> .
<i>canService</i>	Agent→Service	<i>canService(a,s)</i>	Agent <i>a</i> can do service <i>s</i> .
<i>wantConsequence</i>	Agent→Consequence	<i>wantConsequence(a,c)</i>	Agent <i>a</i> wants consequence <i>c</i> .
<i>commitService</i>	Agent→Service	<i>commitService(a,s)</i>	Agent <i>a</i> commits to do service <i>s</i> .
<i>intendAction</i>	Agent→Action	<i>intendAction(a,act)</i>	Agent <i>a</i> intends to do action <i>act</i> .
<i>intendService</i>	Agent→Service	<i>intendService(a,s)</i>	Agent <i>a</i> intends to do service <i>s</i> .
<i>performAction</i>	Agent→Action	<i>performAction(a,act)</i>	Agent <i>a</i> performs action <i>act</i> .
<i>performService</i>	Agent→Service	<i>performService(a,s)</i>	Agent <i>a</i> performs service <i>s</i> .
<i>requestService</i>	Agent→Service	<i>requestService(a,s)</i>	Agent <i>a</i> requests for service <i>s</i> .
<i>tellConsequence</i>	Agent→Agent	<i>tellConsequence(a,b)</i>	Agent <i>a</i> tells a consequence to agent <i>b</i> .
<i>knowConsequence</i>	Agent→Consequence	<i>knowConsequence(a,c)</i>	Agent <i>a</i> knows consequence <i>c</i> .
<i>sayCompetence</i>	Agent→Service	<i>sayCompetence(a,s)</i>	Agent <i>a</i> says its competence about service <i>s</i> .
<i>achieveGoal</i>	Agent→Goal	<i>achieveGoal(a,g)</i>	Agent <i>a</i> achieves goal <i>g</i> .
<i>hasNumberOfActions</i>	Service→int	<i>hasNumberOfActions(s,n)</i>	Service <i>s</i> has the number <i>n</i> of actions.

2.2.3 信任规则

信任概念和关联需要满足的约束以信任规则的形式记录在信任本体中。服务 Agent 通过信任规则分析潜在交互对象的目标或意图、其言语行为和服务执行情况,从社会认知角度推断信任相关维度——即信任信念。

下面给出这些约束规则。采用 OWL^[16]和 SWRL^[17]中构造子作为表示法,包括:“Associate(?x,?y)”表示 *x* Associate *y*,”≠”表示 differentFrom 关联,“^”表示与,“->”表示推断。规则分为两类:第 1 类基于服务 Agent 认知和行为表现,是通过对服务 Agent 的目标、言语及行为的分析,得到与信任维度的推理相关的基本证据的规则;第 2 类是在上述基本证据的基础上,进一步对服务 Agent 的能力、意图、诚实性、坚定性、可预期性和声誉信念进行推理的规则。具体规则形式及其含义见表 2。

这些规则是服务 Agents 用于推导 5 类信任信念的规则,其中未包含声誉信念,因为它与另外几种信任信念有所不同。声誉是服务 Agents 团体对某一被评价对象的一个公众印象,它根据服务提供者的历史交互记录得到,用一个动态值表示。具体来说,与被评估者直接交互的服务 Agents 将自己的经验相互交流并不断累积,最终获得公众共有的对该交互对象的声誉信念,为未来将要与这个服务 Agent 进行交互的其他服务 Agents 提供参考。我们在后面将给出计算服务 Agent 的声誉信念值的方法。

Table 2 Rules in trust sub-ontology
表 2 信任子本体约束规则及其含义

Basic evidence reasoning rule
Rule 1. $knowConsequence(?a,?c)\wedge tellConsequence(?a,?b)\rightarrow knowConsequence(?b,?c)$ //If agent <i>a</i> knows that <i>c</i> is the consequence of an action, and tells this consequence <i>c</i> to agent <i>b</i> , then <i>b</i> will know consequence <i>c</i> .
Rule 2. $partOf(?act,?s)\wedge bringAbout(?act,?c)\wedge achievedByService(?g,?s)\rightarrow partOf(?c,?g)$ //If action <i>act</i> is part of service <i>s</i> , and this action can bring about consequence <i>c</i> , and goal <i>g</i> is achieved by <i>s</i> , then <i>c</i> is part of <i>g</i> .
Rule 3. $hasGoal(?a,?g)\wedge partOf(?c,?g)\rightarrow wantConsequence(?a,?c)$ //If agent <i>a</i> has goal <i>g</i> , and consequence <i>c</i> is part of <i>g</i> , then <i>a</i> wants <i>c</i> .
Rule 4. $performAction(?a,?act)\wedge bringAbout(?act,?c)\rightarrow knowConsequence(?a,?c)$ //If agent <i>a</i> performs action <i>act</i> , and <i>act</i> can bring about consequence <i>c</i> , then <i>a</i> knows <i>c</i> .
Rule 5. $notPerformAction(?a,?act)\wedge partOf(?act,?s)\rightarrow notPerformService(?a,?s)$ //If agent <i>a</i> does not perform action <i>act</i> , and <i>act</i> is part of service <i>s</i> , then <i>a</i> does not perform <i>s</i> .
Rule 6. $requestService(?a,?s)\wedge commitService(?b,?s)\wedge achievedByService(?g,?s)\wedge notPerformService(?b,?s)\rightarrow notAchieveGoal(?a,?g)$ //If agent <i>a</i> requests service <i>s</i> , agent <i>b</i> commits to provide <i>s</i> , and goal <i>g</i> is achieved by <i>s</i> , but <i>b</i> does not perform <i>s</i> , then <i>a</i> does not achieve <i>g</i> .
Rule 7. $cannotService(?a,?s)\rightarrow notPerformService(?a,?s)$ //If agent <i>a</i> cannot provide service <i>s</i> , then <i>a</i> does not perform <i>s</i> .
Rule 8. $notIntendService(?a,?s)\rightarrow notPerformService(?a,?s)$ //If agent <i>a</i> does not intend to provide service <i>s</i> , then <i>a</i> does not perform <i>s</i> .
Trust belief reasoning rule
Rule 9. $hasResource(?a,?e)\wedge hasKnow-how(?a,?k)\wedge needResource(?act,?e)\wedge needKnow-how(?act,?k)\rightarrow canAction(?a,?act)$ //If agent <i>a</i> has resource <i>e</i> and know-how <i>k</i> of an action <i>act</i> , then <i>a</i> can perform <i>act</i> .
Rule 10. $cannotAction(?a,?act)\wedge partOf(?act,?s)\rightarrow cannotService(?a,?s)$ //If agent <i>a</i> cannot perform an action <i>act</i> which is part of service <i>s</i> , then <i>a</i> cannot perform <i>s</i> .
Rule 11. $canService(?a,?s)\wedge partOf(?act,?s)\rightarrow canAction(?a,?act)$ //If agent <i>a</i> can perform service <i>s</i> , then <i>a</i> can perform the <i>act</i> which is part of <i>s</i> .
Rule 12. $hasNumberOfActions(?s,2)\wedge partOf(?act_1,?s)\wedge partOf(?act_2,?s)\wedge canAction(?a,?act_1)\wedge canAction(?a,?act_2)\wedge ?act_1\neq ?act_2\rightarrow canService(?a,?s)$ //If service <i>s</i> consists of 2 actions called <i>act₁</i> and <i>act₂</i> , and <i>a</i> can perform <i>act₁</i> and <i>act₂</i> , then <i>a</i> can perform <i>s</i> . This rule could extend to: if service <i>s</i> consists of <i>n</i> actions called <i>act₁</i> , <i>act₂</i> , ..., <i>act_n</i> , and <i>a</i> can perform all <i>act₁</i> ... <i>act_n</i> , then <i>a</i> can perform <i>s</i> .
Rule 13. $knowConsequence(?a,?c)\wedge wantConsequence(?a,?c)\wedge bringAbout(?act,?c)\rightarrow intendAction(?a,?act)$ //If agent <i>a</i> knows and wants consequence <i>c</i> , but action <i>act</i> can bring about <i>c</i> , then <i>a</i> intends to perform <i>act</i> .
Rule 14. $hasGoal(?a,?g)\wedge achievedByService(?g,?s)\rightarrow intendService(?a,?s)$ //If agent <i>a</i> has goal <i>g</i> , and <i>g</i> is achieved by service <i>s</i> , then <i>a</i> intends to perform <i>s</i> .
Rule 15. $intendService(?a,?s)\wedge partOf(?act,?s)\rightarrow intendAction(?a,?act)$ //If agent <i>a</i> intends to perform service <i>s</i> , then <i>a</i> intends to perform the <i>act</i> which is part of <i>s</i> .
Rule 16. $commitService(?a,?s)\rightarrow intendService(?a,?s)$ //If agent <i>a</i> commits to provide service <i>s</i> , then <i>a</i> intends to perform <i>s</i> .
Rule 17. $hasNumberOfActions(?s,2)\wedge partOf(?act_1,?s)\wedge partOf(?act_2,?s)\wedge intendAction(?a,?act_1)\wedge intendAction(?a,?act_2)\wedge ?act_1\neq ?act_2\rightarrow intendService(?a,?s)$ //If service <i>s</i> consists of 2 actions called <i>act₁</i> and <i>act₂</i> , and <i>a</i> intends to perform <i>act₁</i> and <i>act₂</i> , then <i>a</i> intends to perform <i>s</i> . This rule could extend to: if service <i>s</i> consists of <i>n</i> actions called <i>act₁</i> ... <i>act_n</i> , and <i>a</i> intends to perform <i>act₁</i> ... <i>act_n</i> , then <i>a</i> intends to perform <i>s</i> .
Rule 18. $requestService(?a,?s)\wedge commitService(?b,?s)\wedge cannotAction(?b,?act)\wedge partOf(?act,?s)\rightarrow hasnotIntegrity(?a,?b)$ //If agent <i>a</i> requests service <i>s</i> , and agent <i>b</i> commits to perform <i>s</i> , but <i>b</i> cannot perform the action <i>act</i> which is part of <i>s</i> , then <i>a</i> thinks <i>b</i> is not honest.
Rule 19. $requestService(?a,?s)\wedge sayCompetence(?b,?s)\wedge cannotService(?b,?s)\rightarrow hasnotIntegrity(?a,?b)$ //If agent <i>a</i> requests service <i>s</i> , and agent <i>b</i> says he can perform <i>s</i> , but <i>b</i> cannot perform <i>s</i> , then <i>a</i> thinks <i>b</i> is not honest.
Rule 20. $requestService(?a,?s)\wedge canService(?b,?s)\wedge commitService(?b,?s)\wedge notPerformService(?b,?s)\rightarrow hasnotPersistence(?a,?b)$ //If agent <i>a</i> requests service <i>s</i> , and agent <i>b</i> can and commits to perform <i>s</i> , but <i>b</i> does not perform <i>s</i> , then <i>a</i> thinks <i>b</i> is not persistent.
Rule 21. $requestService(?a,?s)\wedge hasGoal(?b,?g)\wedge achievedByService(?g,?s)\wedge commitService(?b,?s)\wedge notPerformService(?b,?s)\rightarrow hasnotPersistence(?a,?b)$ //If agent <i>a</i> requests service <i>s</i> , and agent <i>b</i> has goal <i>g</i> about achieving <i>s</i> , and commits to perform <i>s</i> , but <i>b</i> does not perform <i>s</i> , then <i>a</i> thinks <i>b</i> is not persistent about his goal.
Rule 22. $requestService(?a,?s_1)\wedge commitService(?b,?s_1)\wedge achievedByService(?g_1,?s_1)\wedge achieveGoal(?a,?g_1)\wedge requestService(?a,?s_2)\wedge commitService(?b,?s_2)\wedge achievedByService(?g_2,?s_2)\wedge notAchieveGoal(?a,?g_2)\wedge ?s_1\neq ?s_2\rightarrow hasnotPredictability(?a,?b)$ //If agent <i>a</i> requests service <i>s₁</i> to achieve goal <i>g₁</i> , and agent <i>b</i> succeeds in performing <i>s₁</i> and helps <i>a</i> to achieve <i>g₁</i> , and agent <i>a</i> requests service <i>s₂</i> to achieve goal <i>g₂</i> , but agent <i>b</i> does not help <i>a</i> to achieve <i>g₂</i> , then <i>a</i> thinks <i>b</i> is not predictable.

3 服务 Agent 信任证据

在信任本体的支持下,服务 Agent 在信任分析评价过程中对信任知识具有了统一的认识,从外部感知获得的信息也可以根据本体知识的结构记录在其自身证据信息库中。

3.1 信任证据的表示

服务 Agent 的信任证据分为 3 类,记作 $IE=ProE \cup BelE \cup DomE$,其中:

- (1) *ProE*:表示与被评估方交互相关的信息,包含针对关联 *performAction,performService,achieveGoal,commitService,tellConsequence,sayCompetence,wantConsequence,requestService,knowConsequence,hasGoal,hasResource,hasKnow-how* 所成立的事实.
- (2) *BelE*:表示被评估方信任信念值累积信息,包含 *CompetenceValue(canAction,canService),IntentionValue(intendAction,intendService),IntegrityValue,PersistenceValue,PredictabilityValue* 标量的取值.
- (3) *DomE*:表示领域知识,包含针对关联 *needResource,needKnow-how,bringAbout,partOf,achievedByService,hasNumberOfActions* 所成立的事实.

3.2 信任证据的获取方式

服务 Agent 的信任证据通过 3 种方式收集,即直接交互方式、观察证据方式和基于大样本源的询问方式.

直接交互方式:服务 Agent 在获取信任证据的过程中首先考虑自身直接交互经验所得的证据,并认为这类证据最为可信.由直接交互的方式获得的交互经验又包含服务 Agent 执行经验和言语行为^[18,19]经验两类.由服务 Agent 执行的方式可以获得 *performAction,performService,achieveGoal,commitService* 等信任关联的事实.由言语交互的方式可以获得 *tellConsequence,sayCompetence,wantConsequence,requestService,knowConsequence* 等信任关联的事实.

观察的方式:在不具有与被评估对象以往交互经验的时候,服务 Agent 可以通过观察对方自身描述以及观察服务所处领域的描述的方式获取推理信息,这部分信息也可直接用于信任分析与推理,而不再对此类信息做处理.由服务 Agent 通过观察方式可以获得 *hasGoal,hasResource,hasKnow-how,needResource,needKnow-how,bringAbout,partOf,achievedByService,hasNumberOfActions,differentFrom* 等信任关联的事实.

基于大样本源的询问方式:在 Internet 开放动态的环境下,参与者可以随意地进入或退出系统,使得服务 Agents 在选择可信服务的时候通常不具有与被评估对象直接交互的经验,并且也缺乏观察证据.我们提出另外一种间接获取信任证据的方式,即基于大样本源的询问方式.这种获取方式的基本思想来自于概率统计中的大样本理论,也就是说,在样本足够大的情况下,多数人认可的结论是可以相信的.基于大样本源的询问获取信任证据的方式为服务 Agent 向足够多(由评估者偏好决定)参与者询问有关被评估对象的某一信任信息(取值为 {0,1}),如果绝大多数的询问对象给出了统一的答案,那么这个答案就是服务 Agent 从大样本源询问方式获得的信息取值.

4 信任值计算

通过信任推理规则可以对已有证据信息进行分析得到信任信念推理结果,服务 Agent 还需要用一组计算规则根据推理结果获得最终信任值.信任值的计算包含两个步骤:第 1 步分别计算各信任维度的值,第 2 步根据服务 Agent 的偏好计算综合信任值.

4.1 信任维度计算

根据信任信念的不同特性,信任维度的计算规则分为两类:

第 1 类针对与服务相关的信念,包括 *Competence* 信念和 *Intention* 信念.服务请求方对提供方在这两个信念上的取值非真即假,没有程度上的区别.根据信任信念推理规则,我们有:如果服务请求方认为服务提供方没有能力提供所需服务,则请求方对提供方的 *Competence* 信念值为 0;反之,则 *Competence* 信念值为 1.*Intention* 的评估方法类似.针对这类信念的计算规则我们称为 I 型信任信念计算规则.

I 型信任信念计算规则. 设 A_i 为该服务相关信念在第 i 次更新后的值,那么,

$$A_i = \begin{cases} 1, & \text{如果第}i\text{次交互之前由信任推理规则得到}canService/intendService = 1 \\ 0, & \text{如果第}i\text{次交互之前由信任推理规则得到}canService/intendService = 0 \end{cases}$$

4.3 数值模拟

为了验证信任计算策略的合理性,我们进行数值实验.该实验分为两部分:第 1 部分验证 II 型信任信念计算规则的合理性;第 2 部分验证信任值评估方法的合理性,同时也用来说明该方法能够有效地帮助服务 Agent 进行服务选择.

服务请求方利用 II 型信任信念计算规则对一个服务提供方的声誉值进行计算,图 5 给出了该服务提供方已有的 50 次交互行为的累计声誉评价结果和所对应的每次交互的更新值.

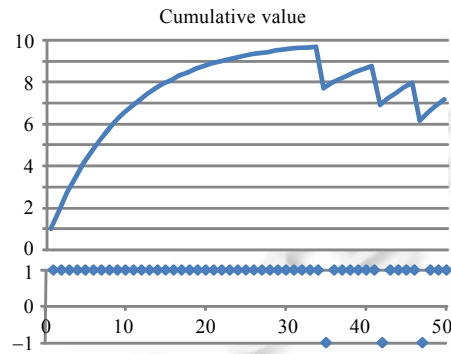


Fig.5 Reputation value of a provider

图 5 服务提供方的声誉评价结果

从图中可以看出,由 II 型信任信念计算规则得到的累计值有以下性质:(1) 该累计值随成功的交互次数增多逐渐趋向于 10,也就是说,无论该提供方曾经交互过多少次,信念累计值都是在一个区间以内变化的,因此方便对经历的交互次数不同的参与者进行比较.(2) 该值的曲线具有慢升快降的性质,即一次的失败经验会导致该累计值的急剧减少(如图中第 35 次交互失败导致其声誉值曲线有显著的下降),而之后的补救则相对困难(如图中在第 36 次到 41 次均成功交互的情况下,其声誉值仍无法恢复到失败之前的大小).这个性质符合人们判断交互对象是否可信的一般常识,即“摧毁容易建立难”的特性.(3) 该累计值越大,服务提供方为了补救一次失败所需的成功次数越多;而累计值越小,其为弥补一次失败所需的成功次数相对较少.这也符合信任评价的一般常识,即交易对象越可信,其一次的失败对其影响相对越大.(4) 该信任值在不断增大的过程中其增长速度逐渐变缓,即信任值越小的情况下,一次的成功经验对该累计值的生长有明显帮助.该特性表明,我们提出的信任信念计算规则能够激励服务提供方,使其交互行为更加可信.

为了验证我们的信任值评估方法所得到的结果符合人们判断信任的一般标准,我们设置了服务 Agent 之间基于信任值进行服务选择的数值模拟场景,分为 5 种不同情况,并依次对所得结果加以分析和讨论.

实验场景 1:服务请求方对 5 个完全相同的服务提供方进行信任评估,重复该服务选择 20 次,所得结果如图 6 所示,其中,横轴为重复实验序号,纵轴为 5 个 Agent 的编号.从图中可见,5 个完全等价的服务被随机选中作为最终交易对象,所得信任信念值和信任值评估结果见表 3.

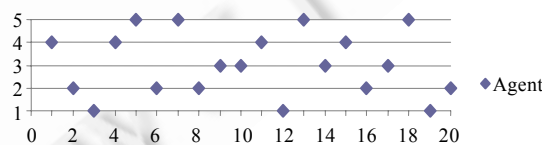


Fig.6 Results of service selection in simulation 1

图 6 实验 1 的服务选择结果

Table 3 Trust values of the providers in simulation 1

表 3 实验 1 的服务提供 Agents 信任评估值

	Com	Inten	Integ	Per	Pre	Rep	Trust
Weight	/	/	0.3	0.2	0.2	0.3	/
1	1	1	7	7	7	8	7.3
2	1	1	7	7	7	8	7.3
3	1	1	7	7	7	8	7.3
4	1	1	7	7	7	8	7.3
5	1	1	7	7	7	8	7.3

实验场景 2:在实验 1 的基础上对 Agent 1 的服务功能施加一定干扰,其他 4 个 Agent 保持不变;重复在这 5 个 Agent 之中选择可信提供方的实验,实验结果如图 7 所示.从图中可以看出,在这种情形下,请求方完全排除 Agent 1,只在 Agent 2 到 Agent 5 之间随机进行服务选择.实验运行过程中的各 Agent 的信任评价结果见表 4.

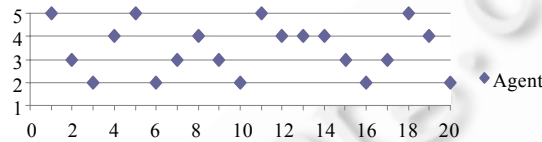


Fig.7 Results of service selection in simulation 2

图 7 实验 2 的服务选择结果

Table 4 Trust values of the providers in simulation 2

表 4 实验 2 的服务提供 Agents 信任评估值

	Com	Inten	Integ	Per	Pre	Rep	Trust
Weight	/	/	0.3	0.2	0.2	0.3	/
1	0	1	7	7	7	8	$-\infty$
2	1	1	7	7	7	8	7.3
3	1	1	7	7	7	8	7.3
4	1	1	7	7	7	8	7.3
5	1	1	7	7	7	8	7.3

实验场景 3:在实验 2 的基础上对 Agent 2 的目标集施加干扰,其余 Agent 保持不变;重复实验,结果如图 8 所示.请求方只在 Agent 3 到 Agent 5 之中随机进行服务选择,信任评估值见表 5.

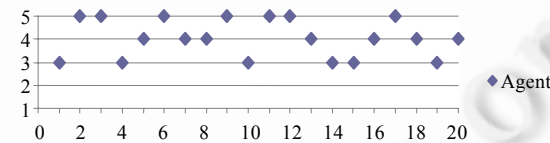


Fig.8 Results of service selection in simulation 3

图 8 实验 3 的服务选择结果

Table 5 Trust values of the providers in simulation 3

表 5 实验 3 的服务提供 Agents 信任评估值

	Com	Inten	Integ	Per	Pre	Rep	Trust
Weight	/	/	0.3	0.2	0.2	0.3	/
1	0	1	7	7	7	8	$-\infty$
2	1	0	7	7	7	8	$-\infty$
3	1	1	7	7	7	8	7.3
4	1	1	7	7	7	8	7.3
5	1	1	7	7	7	8	7.3

实验场景 4:在实验 3 的基础上对 Agent 3 加以设置,使其在与请求方通信时未据实以告.运行结果如图 9 所示,Agent 3 将无法被选中作为交易对象.表 6 给出了信任评估各项的数值结果.

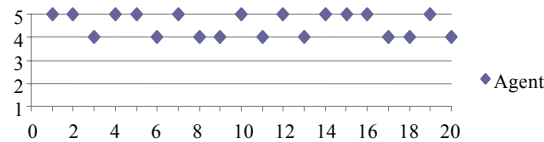


Fig.9 Results of service selection in simulation 4

图 9 实验 4 的服务选择结果

Table 6 Trust values of the providers in simulation 4**表 6** 实验 4 的服务提供 Agents 信任评估值

	Com	Inten	Integ	Per	Pre	Rep	Trust
Weight	/	/	0.3	0.2	0.2	0.3	/
1	0	1	7	7	7	8	$-\infty$
2	1	0	7	7	7	8	$-\infty$
3	1	1	6	7	7	8	7.1
4	1	1	7	7	7	8	7.3
5	1	1	7	7	7	8	7.3

实验场景 5:在实验 4 的基础上对 Agent 3 加以设置,使 Agent 3 前一次与其他 Agent 成功交互并达到请求方目标,之后再次重复我们的实验.实验结果如图 10 显示,只有 Agent 3 会被选中,它们的信任评估结果见表 7.该实验结果说明,虽然请求方认为 Agent 3 的诚实信念值比其他 Agents 略低,但是由于其对于声誉信念的偏重,使得 Agent 3 在综合信任评估以后仍然会被选中.

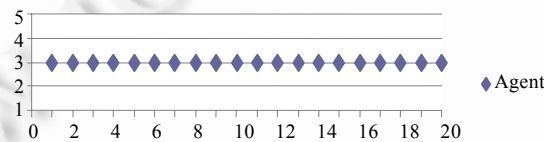


Fig.10 Results of service selection in simulation 5

图 10 实验 5 的服务选择结果

Table 7 Trust values of the providers in simulation 5**表 7** 实验 5 的服务提供 Agents 信任评估值

	Com	Inten	Integ	Per	Pre	Rep	Trust
Weight	/	/	0.3	0.2	0.2	0.3	/
1	0	1	7	7	7	8	$-\infty$
2	1	0	7	7	7	8	$-\infty$
3	1	1	6	7	7	9	7.4
4	1	1	7	7	7	8	7.3
5	1	1	7	7	7	8	7.3

根据以上实验结果,信任值由信任维度值和请求者偏好共同决定,这是因为不同的服务请求方必然具有自己对信任的主观认识.而我们给出的信任维度为请求方提供了信任判断的基础和依据,服务请求方可以根据自己对信任维度的偏好得到信任判断选择结果.实验结果验证了我们在本文中提出的信任建模评估方法的合理性.

4.4 工具实现与案例研究

基于 Protégé^[20]本体开发工具和 JADE(Java agent development framework)^[21]平台,我们用 Java 语言开发了一个服务 Agent 信任评估选择工具.Protégé 用于建立并管理信任本体,并产生 OWL 表示的信任本体.JADE 平台作为服务 Agent 实现平台,使得服务 Agent 能够根据 Protégé 的 OWL 文档及证据事实计算被评估对象的信任值并做出选择决策.下面通过一个网上购物案例对我们提出的信任评估过程的运行结果进行工具展示.

4.4.1 场景描述及信任评估过程

用户 Peter(p)在 Web 上要找一个好的 mp3 搜索引擎下载歌曲.当前存在 4 个搜索引擎提供商 Alice(a), Bob(b), Charles(c)和 David(d)都宣称能够以同样的价钱提供搜索下载歌曲的服务(s).用户 p 需要从中选择一个可信的搜索引擎提供商进行交易.已知用户 p 曾经与 a 提供商和 c 提供商进行过交易,并且前一次与 c 做了一次成功的书籍搜索下载的交易,但与 a 进行的视频搜索交易失败,失败原因是由于 a 实际上不具有用户 p 所需视频文件.通过观察, p 得知 d 其实并不拥有 mp3 文件资源.

因此,用户 p 对 4 个搜索引擎提供商的可信性评估过程包括:(1) 构造下层信任本体;(2) 获取直接交互信任证据及信任信念累计值;(3) 随信任信念进行推理;(4) 更新信任维度值;(5) 计算综合信任值.在计算过程中,用户 p 可以具有不同的偏好,因此可能会选择不同的搜索引擎提供商.

4.4.2 运行结果

图 11 给出了案例描述的基于信任评估的服务选择过程.其中, $r1$ 代表服务请求方, $a1\sim a4$ 代表服务提供方.该过程为 $r1$ 发起服务请求, $a1\sim a4$ 分别做出响应, $r1$ 根据其记录在信任本体中的知识以及证据库中的事实证据对提供方进行信任评估,采用偏好 $\{0.3,0.2,0.2,0.3\}$ 做出决策,实验中, $r1$ 最终选择 $a3$.

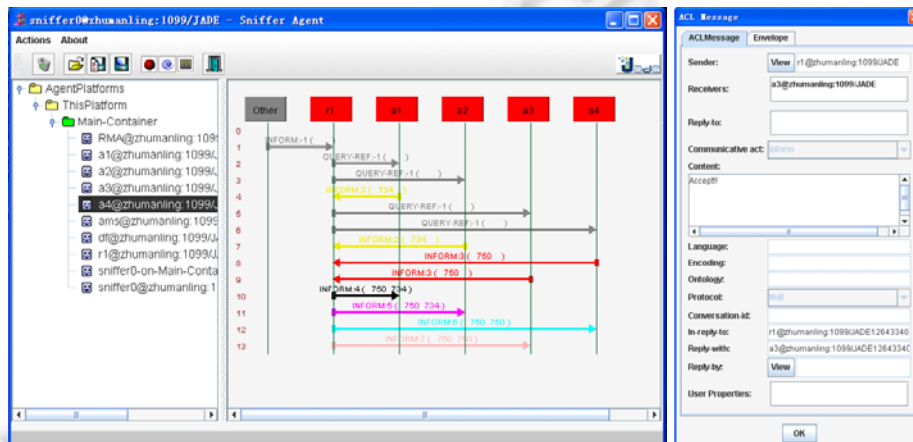


图 11 Service agent selection based on trust evaluation in JADE platform

图 11 在 JADE 平台上基于信任评估的服务选择过程

5 相关工作及讨论

目前,Web 服务选择过程中的可信问题研究一般是以服务质量(QoS)作为服务可信性的评价依据.QoS 包括多个维度,如服务响应时间、服务延迟、服务界面友好性等.Diamadopoulou 等人^[3]提出依靠一个本体来定义 QoS 参数,再根据 QoS 参数值高低的一个权衡来选择可信服务的方法.Maximilien 等人^[22]提出应该将客观质量参数(如响应时间)和主观质量参数(如可用性)加以区分,使得客观质量能够被自动测量,有利于减轻请求方评价的负担,并降低错误评分出现的概率.但是,这类基于 QoS 的方法缺少对服务自主性的考虑,服务在一定程度上体现的是其所有者的目标和意图,单纯根据服务质量对服务可信性进行评价的方法难以对服务提供者的行为做出预期,因此无法对未来服务选择做出指导.

当前,很多相关工作以传统 Agent 系统的信任建模工作为基础,将 Agent 领域的信任决策框架引入服务可信性评估的方法.Beth 模型^[23]采用事件概率的方式来表述和度量信任关系,基于一定的概率分布假设.Mui 等人^[10]的信任模型提出使发起请求的 Agent 在不断交互过程中学习对方的策略,以应对说谎或恶意 Agent 的方法.该方法对 Agents 之间的交互次数有一定要求,而在网络环境下,服务 Agent 可以随时退出系统或改变身份重新进入系统的情况下,该要求较难满足.Jósang 模型^[24]对公共密钥方法进行改进,将密钥和持有者绑定,并考虑是否具

有推荐信任的能力,包含一个用于组合信任和推荐信息的逻辑算子集合,信任可传递,其缺点是无法解决恶意推荐对信任度评估的影响.Huynh 等人^[25]的信任声誉模型 FIRE 的基本思想是让 Agent 相互评分,综合各种信息源计算被评估对象的信任值.这类方法主要关注如何将直接交互经验、第三方推荐信息或权威机构认证等不同的信息源综合计算得到信任值.Castelfranchi 等人^[12]提出从社交认知的角度建立基于评估对象的思维状态的信任模型,认为信任具有若干维度,包括能力、动机等,信任关系的建立需要对各个维度加以考察综合得出.但其工作中并没有给出从已有信息中如何推导信任维度的方法,也缺乏对以往交互历史的考虑,因此也很难将该方法用于实践.

国内的研究也已有不少关于信任度量和评估的工作.文献[26]针对开放网络环境下信任管理的问题,提出了一个主观信任管理模型,运用模糊集合理论对信任管理问题进行建模,给出了信任类型的定义机制和信任的评价机制,并提出了信任关系的推导规则,为开放网络环境中的信任管理研究提供了一个新的思路.文献[27]针对现有的信任模型无法刻画信任关系的动态性的问题,提出了一个适用于网构软件的信任度量及演化模型.该模型对信任关系度量过程和信任信息传递及合并过程进行了合理抽象,而且提供了一种用于促进协同实体间信任关系的自动形成与更新的方法.文献[28]针对 P2P 环境中信任关系难建立、现有模型不能很好地解决模型的迭代收敛性问题,提出了参考人际网络中基于推荐的信任关系建立方法,并给出了基于 P2P 环境的信任模型的数学分析和分布式实现方法,该模型在迭代的收敛性问题上比原有模型有较大改进.文献[29]针对当前基于安全凭证和反馈信息的信任评估方法存在的缺陷,参考社会学的信任模型,利用量化主体能力属性提升信任评估的准确度.

与这些相关工作相比,本文提出的信任建模评估方法具有一定的优势.首先,本文基于社交认知的观点将信任划分为多个不同类型的维度,并分别对这些维度进行了分析和挖掘,建立了这些维度的推理规则.其次,本文所划分的维度不仅包含通过当前的观察推理证据得到的信任信念——能力及意图信念,而且包含考查个体交互历史纪录的诚实、坚守承诺以及可预期信念,同时又包含刻画公众评价信息的声誉信念.本文的信任维度出自社会学领域对信任的研究,因此更为合理,就目前已知的工作而言,本文考虑到的信任维度更全面.最后,本文给出的信任评估方法不仅可推理而且可计算.案例研究表明,本文提出的信任建模方法能够很好地刻画信任评估过程,并得到与一般常识一致的基于信任的服务选择结果.

6 结束语

本文基于社交认知的服务可信性建模评估方法的主要特点包括:(1) 提出了一个基于服务 Agent 的框架,并给出了服务 Agent 的结构,以及完整的服务 Agent 信任评估过程,该过程在开放网络环境下以服务 Agent 的视角对服务可信性进行了研究;(2) 从社交认知的观点建立了一个信任本体,该信任本体规范了信任概念体系,为服务 Agents 之间的信任关系提供了更好地理解,并能够支持信任的分析和推理;(3) 提出一组基于信任本体的服务 Agent 信任计算规则,可以帮助服务 Agents 根据已有信息及自身偏好进行信任评估;(4) 案例研究表明,我们的信任建模评估方法可以有效地帮助服务 Agents 进行服务选择和决策.

基于 Agent 的服务信任建模是开放网络环境下服务可信性分析评估的一种可行而有效的解决方案,是一个值得深入研究的课题,有可能在未来服务计算领域发挥作用.本文进一步的工作包括:优化信任值计算规则,深入分析开放网络环境下具有不同属性类型的服务 Agents(如 good-will 类型即好意的服务 Agent 或 malicious 类型即恶意的服务 Agent 等)在我们的信任评估模型中的长期运行结果;建立组织下的服务 Agent 信任评估模型,包括团体对团体的信任建模分析、团体和另外个体的相互信任评价及团体的信任评估结果对团体中个体可信性的影响等.

References:

- [1] Wang Y, Vassileva J. A review on trust and reputation for Web service selection. In: Proc. of the 27th Int'l Conf. on Distributed Computing Systems Workshops. 2007. 25. [doi: 10.1109/ICDCSW.2007.16]

- [2] Vu LH, Hauswirth M, Aberer K. QoS-Based service selection and ranking with trust and reputation management. In: Proc. of the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE. LNCS 3760, 2005. 466–483.
- [3] Diamadopoulou V, Makris C, Panagis Y, Sakkopoulos E. Techniques to support web service selection and consumption with QoS characteristics. *Journal of Network and Computer Applications*, 2008,31(2):108–130. [doi: 10.1016/j.jnca.2006.03.002]
- [4] Maximilien EM, Singh MP. A framework and ontology for dynamic Web services selection. *IEEE Internet Computing*, 2004,8(5): 84–93. [doi: 10.1109/MIC.2004.27]
- [5] Hendler J. Agents and the semantic Web. *IEEE Intelligent Systems*, 2001,16(2):30–37. [doi: 10.1109/5254.920597]
- [6] Huhns MN. Agents as Web services. *IEEE Internet Computing*, 2002,6(4):93–95. [doi: 10.1109/MIC.2002.1020332]
- [7] Huhns MN, Singh MP. Service-Oriented computing: Key concepts and principles. *IEEE Internet Computing*, 2005,9(1):75–81. [doi: 10.1109/MIC.2005.21]
- [8] Sycara K, Paolucci M, Soudry J, Srinivasan N. Dynamic discovery and coordination of agent-based semantic Web services. *IEEE Internet Computing*, 2004,8(3):66–73. [doi: 10.1109/MIC.2004.1297276]
- [9] Liu L, Liu Q, Chi C, Jin Z, Yu E. Towards a service requirements ontology based on knowledge and intention. In: Proc. of the 6th Int'l Conf. on Quality Software. 2006. 452–462. [doi: 10.1109/QSIC.2006.65]
- [10] Mui L, Mohtashemi M, Halberstadt A. A computational model of trust and reputation. In: Proc. of the 35th Hawaii Int'l Conf. on System Science (HICSS-35 2002). IEEE Computer Society, 2002. 2431–2439. [doi: 10.1109/HICSS.2002.994181]
- [11] Sabater J, Sierra C. REGRET: A reputation model for gregarious societies. In: Castelfranchi C, Johnson L, eds. Proc. of the 1st Int'l Joint Conf. on Autonomous Agents and Multi-Agent Systems. 2002. 475–482.
- [12] Castelfranchi C, Falcone R. Social trust: A cognitive approach. Castelfranchi C, Tan YH, eds. Proc. of the Trust and Deception in Virtual Societies. Kluwer Academic, 2000. 55–90.
- [13] McKnight DH, Chervany NL. What is trust? A conceptual analysis and an interdisciplinary model. In: Proc. of the AIS. 2000. 827–833.
- [14] Ryutov T. A socio-cognitive approach to modeling policies in open environments. In: Proc. of the 8th IEEE Int'l Workshop, Policies for Distributed Systems and Networks (POLICY 2007). 2007, 29–38. [doi: 10.1109/POLICY.2007.3]
- [15] Zhu ML, Jin Z. A trust measurement mechanism for service agents. In: Proc. of the IEEE/WIC/ACM Int'l Joint Conf. on Web Intelligence and Intelligent Agent Technology (WI-IAT 2009). 2009. 375–382. [doi: 10.1109/WI-IAT.2009.66]
- [16] OWL. 2004. <http://www.w3.org/TR/owl-ref/#EnumeratedClass/>
- [17] SWRL. 2004. <http://www.w3.org/Submission/SWRL/>
- [18] Austin JL. *How to Do Things with Words*. Cambridge: Harvard University Press, 1962.
- [19] Searle JR. *Speech Acts: An Essay in the Philosophy of Language*. Cambridge: Cambridge University Press, 1969.
- [20] Protégé. <http://protege.stanford.edu/>
- [21] JADE. <http://jade.tilab.com/>
- [22] Maximilien EM, Singh MP. Toward autonomic Web services trust and selection. In: Proc. of the 2nd Int'l Conf. on Service Oriented Computing. 2004. 212–221. [doi: 10.1145/1035167.1035198]
- [23] Beth T, Borcherding M, Klein B. Valuation of trust in open network. In: Gollmann D, ed. Proc. of the European Symp. on Research in Security (ESORICS). Brighton: Springer-Verlag, 1994. 3–18. [doi: 10.1007/3-540-58618-0_53]
- [24] Jøsang A, Gray E, Kinatader M. Simplification and analysis of transitive trust networks. *Web Intelligence and Agent Systems*, 2006,4(2):139–161.
- [25] Huynh D, Jennings NR, Shadbolt NR. Developing an integrated trust and reputation model for open multi-agent systems. In: Proc. of the 7th Int'l Workshop on Trust in Agent Societies. New York: ACM Press, 2004. 65–74.
- [26] Tang W, Chen Z. Research of subjective trust management model based on the fuzzy set theory. *Journal of Software*, 2003,14(8): 1401–1408 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/14/1401.htm>
- [27] Wang Y, Lü J, Xu F, Zhang L. A trust measurement and evolution model for internetware. *Journal of Software*, 2006,17(4): 682–690 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/17/682.htm> [doi: 10.1360/jos170682]
- [28] Dou W, Wang HM, Jia Y, Zou P. A recommendation-based peer-to-peer trust model. *Journal of Software*, 2004,15(4):571–583 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/15/571.htm>

- [29] Li HH, Du XY, Tian X. A capability enhanced trust evaluation model for web services. Chinese Journal of Computers, 2008,31(8): 1471-1477 (in Chinese with English abstract).

附中文参考文献:

- [26] 唐文,陈钟.基于模糊集合理论的主观信任管理模型研究.软件学报,2003,14(8):1401-1408. <http://www.jos.org.cn/1000-9825/14/1401.htm>
- [27] 王远,吕建,徐锋,张林.一个适用于网构软件的信任度量及演化模型.软件学报,2006,17(4):682-690. <http://www.jos.org.cn/1000-9825/17/682.htm> [doi: 10.1360/jos170682]
- [28] 窦文,王怀民,贾焰,邹鹏.构造基于推荐的 Peer-to-Peer 环境下的 Trust 模型.软件学报,2004,15(4):571-583. <http://www.jos.org.cn/1000-9825/15/571.htm>
- [29] 李海华,杜小勇,田莹.一种能力属性增强的 Web 服务信任评估模型.计算机学报,2008,31(8):1471-1477.



朱曼玲(1981—),女,安徽休宁人,博士,主要研究领域为软件工程,知识工程,可信性建模.



金芝(1962—),女,博士,教授,博士生导师,CCF 高级会员,主要研究领域为软件需求工程,领域建模,基于知识的软件工程.