

Web 软件可信性的模糊评价方法^{*}

赵玉洁, 罗新星

(中南大学商学院, 长沙 410083)

摘要: 在对 ISO/IEC 9126 模型及 Web 软件属性图进行改进和扩充后,应用因子分析法构建了 Web 软件可信性评价指标体系,运用结构熵值法确定了各个可信指标的权重,并运用改进的证据合成方法构造专家评价信息的模糊评价矩阵;最后,运用置信度识别准则对软件可信性进行等级评价和得分计算,对应可信性决策规则集得出相应的策略。结合对 Web 软件中网上银行项目的实例分析说明了该方法的实用性和有效性。

关键词: Web 软件可信性; 结构熵值法; 改进证据合成理论; 置信度识别准则; 可信性决策规则集

中图分类号: TP311

文献标志码: A

文章编号: 1001-3695(2014)04-1072-05

doi:10.3969/j.issn.1001-3695.2014.04.028

Fuzzy evaluation method for Web software trustworthiness

ZHAO Yu-jie, LUO Xin-xing

(Dept. of Business, Center South University, Changsha 410083, China)

Abstract: Firstly, this paper improved and expanded ISO/IEC 9126 model and Web software attribute graph to build the Web software trustworthiness evaluation indicator system by using factorial analysis. Secondly, it used the structure entropy value method to determine all the trustworthiness indicators' weight, and used the improved synthetic method to construct fuzzy evaluation matrix from expert evaluation information. Finally, it used confidence recognition criteria to assess the rank and calculate score for first-level trustworthiness indicator and integral trustworthiness, and made a decision from corresponding trustworthiness decision rule set. With the example of online banking project in Web software analysis it illustrates the method's practicability and validity.

Key words: Web software trustworthiness; structure entropy method; improved D-S evidence theory; confidence recognition criteria; trustworthiness decision rule set

0 引言

互联网的普及和网络技术的迅猛发展,使得软件系统逐步走向动态开放、复杂多变的环境中。人们对软件的可信性产生了迫切的要求,并在国内外对可信软件开展了大量的研究。美国《国家软件发展战略(2006—2015年)》将开发高可信软件放在首位,并提出了下一代软件工程的构想;我国国家自然科学基金委员会“十一五”期间重大研究计划——“可信软件基础研究”重大研究计划也于2008年正式启动。

软件可信评估问题是可信软件领域中的一个重要问题,它涉及软件一系列定性和定量可信指标。早期研究软件可信是对单维可信属性进行分析、测试和验证的,但是随着软件应用环境越来越复杂,在评估一个软件的可信性时,需要综合考虑多维可信属性。对于这些可信属性,并不是一视同仁的。各可信属性的重要程度取决于系统的应用需求、设计、开发成本和外在环境等多方面的因素,因此对软件可信性的度量和评价实际上是一个综合评价的过程。由于软件可信性是由可靠性、安全性、可用性等多个可信属性组成,软件可信性评估可以看做是一种多属性决策分析(multiple attribute decision analysis, MADA)问题。近年来,结合软件应用实际和领域专业背景,许多

应用于考察一个或多个可信属性的可信性评估模型或方法被提出,其中具有代表性的包括统一可信性模型(UMD)^[1]、Bayesian 网模型^[2]、基于本体的集成评估方法^[3]、软件多维度评估 Web 模型^[4]、Markov 用例模型^[5]、基于模糊理论的多属性评估方法^[6]、可信度估算模型^[7]、自适应可信性模型^[8]和开源构件可信性评估方法^[9]等。为了满足具有特定可信需求的软件可信性评估问题,如航空飞行控制软件、智能控制软件等,许多仅能度量或评估单一可信属性的模型也相继被提出。

1 Web 软件可信性评价指标体系

1.1 软件可信性

软件可信性涉及航空航天、原子核反应堆、军事科学、国防建设以及金融、证券、医疗卫生、汽车、电力等国民经济的各个高科技领域,它已经直接牵涉到人们的生活质量,甚至关系到人类生命、财产的安全问题。“可信”一词最初引入计算机领域时使用的是 dependability,也就是软件可信性问题,随着计算机网络技术的诞生,开始使用 trustworthiness 一词来描述软件可信性问题。Trustworthiness 不仅包括了最初的 dependability,并且是传统软件质量概念的延伸,它更关注综合层面的软件可信性属性和从整体角度上考虑如何度量软件可信性。

收稿日期: 2013-07-03; 修回日期: 2013-08-25
(71221061)

基金项目: 国家自然科学基金资助项目(90818014); 国家自科基金委创新群体项目

作者简介: 赵玉洁(1986-),女,甘肃兰州人,博士研究生,主要研究方向为可信软件(yuanyuan860831@163.com); 罗新星(1956-),男,湖南邵阳人,教授,博导,主要研究方向为管理信息系统。

软件可信性 (software trustworthiness) 是指软件的行为和结果符合用户预期,并在受到干扰时仍能提供连续服务的能力^[10]。软件可信性是对软件行为的整体刻画,是对软件可靠性、安全性、正确性、时效性等一系列特征属性的综合度量,是软件可信基础研究的核心问题之一。软件可信性度量的问题其实就是软件可信性的属性选取和度量、用户预期的表达和依据对被选择属性的度量值和用户预期对两者符合性的量化问题。

1.2 Web 软件可信性评价指标体系的建立

Web 软件是在互联网技术快速发展背景下,软件形态向网络化方向转变趋势下诞生的新型软件系统,Web 软件正以其开放性等特性成为互联网资源、技术与方法聚合的完美结合与体现的优秀典范,是可信软件中极具代表性的一类软件。但同时由于其运行环境的不确定性与复杂性,Web 软件可信性也是 Web 软件产品成功的关键因素,所以,对 Web 软件可信性进行可信性的评估是亟待解决的问题。

针对上述问题,在 ISO/IEC 9126 模型的基础上,结合 Behkamal 等人^[11]提出的 Web 软件属性图,再充分考虑到 Web 软件主要关注点为外部使用满意度和用户个人信息的保密度,将原有模型中的 12 个二级可信属性扩充为 18 个,扩充的 6 个属性分别为 RSA 加密、DES 加密、可追踪性、可访问性、可定制性及可导航性,并将安全性提升为一级属性,进而通过对 Web 软件可信性评估问题编制问卷。此次问卷的测量尺度采用 Likert 五级量表,分别从非常重要、重要、一般、不重要到非常重要给予 5、4、3、2、1 的分值(共发放问卷 200 份,回收的有效问卷数量为 176 份,方法对象为相关领域的专家和学者),对问卷所得数据通过因子分析法进行计算,得出各可信性属性的因子载荷与累计贡献率,得到了更具解释能力的 Web 软件可信性指标体系,如表 1 所示。

2 Web 软件可信性的模糊评价

模糊综合评价方法是由 Azdah 提出,对于解决具有多种属性、考虑多种因素影响的问题,运用模糊数学的工具对事物进行全面综合的评价方法。该方法的关键之处有两点:a)确定指标体系中各级指标的权重;b)构建关于各级指标的模糊评价矩阵。以下将分别介绍 Web 软件可信性评价中指标权重和模糊评价矩阵的确定方法。

2.1 Web 软件可信性属性权重分配

本文采用结构熵值法来确定 Web 软件可信性属性的权重,该方法的优点是:将专家意见的德尔菲调查法和模糊分析方法相结合、定性定量分析相结合,充分发挥了主客观赋权法的优势,对各指标进行重要性排序也相较于其他主观赋权法更具可操作性和简便性。结构熵权法的步骤是:a)根据已经建立的 Web 软件可信性指标体系,由专家对于给定属性进行德尔菲法的定性判断,通过多轮反复形成对各个可信属性指标的重要性排序;b)对专家定性判断的不确定性采用熵值法定量计算隶属度,并进行模糊度分析,引进专家认知盲度消除数据噪声;c)通过多轮计算对软件可信性属性权重进行分配。运用该方法来确定 Web 软件可信性评价指标的权重,既可以避免大样本客观数据搜集的难度,专家根据实际经验对指标进行重要性排序更易理解和易操作,又可以通过模糊分析降低指标

权重确定过程中存在的主观性问题,从而提高 Web 软件可信性评价指标权重确定工作的效率和权重计算结果的可靠性。

表 1 Web 软件可信性评价指标体系

一级指标	二级指标	指标说明
可靠性	容错性	度量 Web 软件防止外部接口错误扩散而导致系统失效的能力(主要是对外错误的隔离)
	成熟性	度量 Web 软件为避免软件内部的错误扩散而导致系统失效的能力(主要是对内错误的隔离)
	易恢复性	度量 Web 软件在出现错误的情况下重新恢复原有的功能和性能的能力
	易访问性	度量 Web 软件在需要时间内能否及时提供指定服务的能力
可用性	易理解性	度量 Web 软件提供给用户的信息是否清晰、准确和易懂,使用户能够快速理解的能力
	易学性	度量 Web 软件使用户能学习其应用的能力
	易操作性	度量 Web 软件产品使用户能易于操作和控制它的能力
	可定制性	度量 Web 软件能否根据用户的需求添加服务以提高用户满意度的能力
功能性	可导航性	度量 Web 软件能否满足用户的便利性提供有效信息的能力
	准确性	度量 Web 软件提供给用户功能的精确度是否符合目标的能力
	互操作性	度量 Web 软件在与其他系统进行交互时对信息加以利用的能力
	可追踪性	度量 Web 软件在不同平台上正确处理信息、追踪系统操作的能力
效率	时间性能	度量 Web 软件处理特定的业务请求所需要的响应速度的快慢
	资源利用性	度量 Web 软件处理特定业务请求所消耗的系统资源的大小
安全性	RSA 加密	度量 Web 软件运行于网络环境时的信息安全性,主要是权限问题
	DES 加密	度量 Web 软件运行于网络环境时的数据安全性,主要是密码问题

2.1.1 对可信属性进行重要性排序的定性分析

利用专家的实际工作经验对给定的 Web 软件给出可信性属性重要性的一个排列顺序。首先根据设计重要性排序方法规定的程序和要求,向熟悉 Web 软件的若干个专家进行问卷调查;通过征询和反馈,最终形成的专家排序意见即为软件可信性属性的重要性排序,如表 2 所示。设有 k 个专家参加软件可信性属性重要性排序,得到 k 张重要性排序表,每一张对应一组重要性排序,记为 $U = (u_1, u_2, \dots, u_n)$,每一组重要性排序集对应的数组为 $(a_{i1}, a_{i2}, \dots, a_{in})$ 。 k 个专家意见得到的软件可信性属性重要性排序矩阵记为 $A(A = (a_{ij})_{k \times n}, i = 1, 2, \dots, k; j = 1, 2, \dots, n)$,其中 a_{ij} 表示第 i 个专家对第 j 个可信性属性的评价。

2.1.2 对重要性排序的不确定性进行定量分析

重要性排序结果具有模糊性,利用熵理论计算其隶属度,对重要性排序的不确定性定量分析,对可能产生潜在偏差的数据进行处理。利用信息熵这个工具计算出各个软件可信性属性重要性排序的变异程度。

重要性排序的隶属度函数为

$$\chi(I) = -\lambda p_n(I) \ln p_n(I) \quad (1)$$

令 $p_n(I) = \frac{m-I}{m-1}$, 取 $\lambda = \frac{1}{\ln(m-1)}$, 将 λ 代入式(1)得

$$\chi(I) = -\frac{1}{m(m-1)} \left(\frac{m-I}{m-1} \right) \ln \left(\frac{m-I}{m-1} \right) \quad (2)$$

化简得到

$$\chi(I) = -\frac{m-I}{m-1} \times \frac{\ln(m-I)}{\ln(m-1)} + \frac{m-I}{m-1} \quad (3)$$

令 $\mu(I) = \frac{\chi(I)}{m-1}$, 代入式(3)得

$$\mu(I) = -\frac{\ln(m-I)}{\ln(m-1)} \quad 0 < \mu < 1; I=1, 2, \dots, j+1 \quad (4)$$

其中, I 是专家按照重要性排序对各个软件可信性属性给出的定性排序数, 并将其对应于具体整数分值, 则上述分值在区间 $[1, n]$ 内取值。若认为某个可信性属性处于第一选择, 则 I 取值为 1, 若为第二选择, 则取值为 2, 其他依此类推。如此, 软件可信性属性的重要性由专家相应的评价分值来度量。

2.1.3 对定量转换后的结果进行认识度分析

假设 k 个专家对于软件可信性属性的话语权相同, 即 k 个专家对于可信性属性 u_j 具有一致看法, 称为专家的平均认识度, 记为 b_j , 令

$$b_j = \frac{b_{1j} + b_{2j} + \dots + b_{kj}}{k} \quad (5)$$

定义专家 z_i 对于软件可信性属性由认知产生的不确定性, 称为专家认知盲度, 记为 Q_j , 令

$$Q_j = \frac{|\{ |\max(b_{1j}, b_{2j}, \dots, b_{kj}) - b_j| + |m_i(b_{1j}, b_{2j}, \dots, b_{kj}) - b_j| \}|}{2} \quad (6)$$

对于每一个软件可信性属性 u_j , 定义 k 个专家对于 u_j 的总体认识度为 x_j , 令

$$x_j = b_j(1 - Q_j) \quad (7)$$

对 u_j 进行归一化处理, 记为 a_j , 令 $a_j > 0 (j=1, 2, \dots, n)$, 且 $\sum_{j=1}^n a_j = 1$ 。

集合 $(a_{i1}, a_{i2}, \dots, a_{in})$ 即为 k 个专家对软件可信性属性集 $U = (u_1, u_2, \dots, u_n)$ 的重要性排序的一致性整体判断, $W = (a_{i1}, a_{i2}, \dots, a_{in})$ 即为软件可信性属性的权重向量。

$$\alpha_j = x_j / \sum_{j=1}^n x_j \quad (8)$$

2.2 基于证据理论的模糊评价矩阵构造

2.2.1 改进的证据合成方法

证据理论是一种处理不确定性推理的有效方法, 是由 Dempster 于 1967 年首次提出, 并由他的学生 Shafer 加以改进和推广, 因此也称为 Dempster-Shafer (D-S) 证据理论。它是一种决策理论, 可以很好地处理由于信息表达带来的不确定性和不准确性, 借助证据理论可以把整个问题和证据细分为多个子问题、子证据, 进而根据 D-S 合成准则计算出它们共同作用对每一个证据的支持程度, 对所有的子证据进行融合, 便可以得到整个问题的解。1994 年 Yang 和 Singh 将证据理论用于解决准则值不完整的多准则决策问题, 并解决了建模不完整和信息不精确的问题。总而言之, 证据理论是证据理论与决策理论相结合的产物, 随着模糊数学、概率论和区间计算等相关理论的引入, 该方法已具备了一些明显的优点, 比如能处理具有定性定量准则混合的决策问题、可以处理信息的不完整和不精确、能较好地处理具有模糊性及不确定性信息的合成问题等。因此, 将证据理论用来处理软件可信性评估中存在的问题具有非常良好的应用前景。

本文采用李仕峰等人^[12]提出的改进证据合成方法, 此方法较传统证据理论的优点在于:

a) 在基于证据可信度和证据确定性概念的基础上, 根据证据有效性的定义和度量的方法, 再结合证据冲突全局分配的思想, 改进了证据合成的方法。

b) 改进后的方法提高了在处理证据时的收敛性和鲁棒性, 并根据 Jousselme 等人^[13]提出的证据距离、证据相似度以及证据被支持程度的定义和方法给出了证据可信度的定义, 可以利用证据有效性来区分证据的重要程度, 相比于之前仅用证据可信度来区分证据重要性的模型更为充分合理。

c) 根据证据冲突全局分配的思想对证据合成法则进行修改, 可以进一步提高合成法则的鲁棒性。李仕峰等人^[12]也通过编程计算验证了此方法的收敛性。

2.2.2 Web 软件可信性的模糊评价矩阵

在 Web 软件可信性评估中利用改进后的证据合成方法, 可以根据证据有效性来区分不同可信属性证据的重要程度, 也可以考虑到可信证据冲突全局的分配。同时, 可以更为有效地处理在对 Web 软件进行可信性评价中群组专家意见中不确定性信息的合成问题。利用此方法具有的鲁棒性和收敛性, 可以进一步提高评价结果的聚焦程度, 从而更好地处理异常的评价信息, 进而增强对 Web 软件可信性评级的可识别性和可靠性。

定义 1 设 $m_1, m_2, \dots, m_n$ 为同一识别框架 Θ 下的 n 个证据, 证据 $m_i (i=1, 2, \dots, n)$ 的相对可信度为

$$\text{Crd}(m_i) = \frac{\sup(m_i)}{\max_{1 \leq j \leq n} \sup(m_j)} \quad (9)$$

其中, $\sup(m_i)$ 为证据 m_i 被支持的程度。

利用信息熵理论对证据的确定性进行界定和度量。

定义 2 假设 $m(A) (A \subseteq \Theta)$ 为识别框架 Θ 下的证据 m 的基本可信度分配函数, $|\Theta| = N, 2^\Theta = \{A_1, A_2, \dots, A_{2N}\}$, 则证据 m 所提供的信息的相对确定性为

$$\text{Cer}(m) = 1 - \frac{H(m)}{|H(m)|} \quad (10)$$

其中: $H(m)$ 为度量证据 m 所包含信息不确定性程度的信息熵值, $|H(m)|$ 为信息完全不确定时的信息熵, 分别为

$$H(m) = -\rho \sum_{i=1}^{2N} m(A_i) \ln[m(A_i)] \quad (11)$$

$$|H(m)| = -\rho \sum_{i=1}^{2N} \left(\frac{1}{2^N} \ln \frac{1}{2^N} \right) = -\rho \ln \frac{1}{2^N} \quad (12)$$

其中, $\rho > 0$ 为熵值系数。

定义 3 设 $m_1, m_2, \dots, m_n$ 为同一识别框架 Θ 下的 n 个证据, $\text{Crd}(m_i)$ 和 $\text{Cer}(m_i)$ 分别为证据 $m_i (i=1, 2, \dots, n)$ 的相对可信度和相对确定性, 证据 m_i 的效用 $\text{Uti}(m_i)$ 和相对有效性 $\text{Eff}(m_i)$ 分别为

$$\text{Uti}(m_i) = \text{Crd}(m_i) e^{\text{Crd}(m_i) - 1} \quad (13)$$

$$\text{Eff}(m_i) = \frac{\text{Uti}(m_i)}{\max_{1 \leq j \leq n} [\text{Uti}(m_j)]} \quad (14)$$

利用证据的相对有效性对原始证据模型进行修正:

$$m'_i(A) = \begin{cases} \text{Eff}(m_i) \times m_i(A) & A \neq \Theta \\ 1 - \sum_{B \subseteq \Theta} \text{Eff}(m_i) \times m_i(B) & A = \Theta \end{cases} \quad (15)$$

利用证据效用值确定证据冲突分配权重 $w(A, m)$:

$$w(A, m) = \frac{\sum_{i=1}^n \text{Uti}(m_i)}{\sum_{j=1}^n \text{Uti}(m_j)} \times m_i(A) \quad (16)$$

综上, 对证据合成规则进行修改。

定义 4 设 $m_1, m_2, \dots, m_n$ 为同一识别框架 Θ 下的 n 个证据, 证据合成规则为

$$m(A) = \begin{cases} 0 & A = \Phi \\ \sum_{\cap A_i = \Phi} \prod_{1 \leq j \leq n} m'_j(A_i) + K'w(A, m) & A \neq \Phi \end{cases} \quad (17)$$

其中: $A_i (i=1,2,\dots,2^{|\Theta|}) \subset \Theta; K'$ 为修正后证据模型的总冲突:

$$K' = \sum_{\bigcap A_i = \Phi} \prod_{1 \leq j \leq n} m'_j(A_i) \quad (18)$$

其中, $\sum_{A \in \Theta} m(A) = 1$ 。

2.2.3 Web软件可信性的模糊评价矩阵的构造

假设 $k(k \geq 2)$ 为专家对指标 u 进行评价,评语集为 $V = \{v_j (j=1,2,\dots,n)\}$ 。令专家 $l(l=1,2,\dots,k)$ 的评价结果为 $p_l = (v_1:p_l(v_1); v_2:p_l(v_2); \dots; v_n:p_l(v_n))$, $\sum_{j=1}^n p_l(v_j) = 1$ 。其中, $p_l(v_j)$ 表示专家 l 认为对于 u 的评语为 v_j 的可能性程度。将此问题转换为证据模型,则可以表述为: $m_1, m_2, \dots, m_n$ 为同一识别框架 Θ 下的 k 个证据, $2^\Theta = \{A(v_j) | j=1,2,\dots,n\}$, 并且 $m_l(v_j) = p_l(v_j)$ 。按照上述改进的证据合成方法对 k 个证据进行合成,得到 $m(A(v_1)), m(A(v_2)), \dots, m(A(v_n))$ 定义为指标 u 隶属于评语集 $V = \{v_j (j=1,2,\dots,n)\}$ 的隶属度向量。因此,对 Web 软件的 m 种属性,可记为 $U = \{u_1, u_2, \dots, u_m\}$, 评语集为 $V = \{v_1, v_2, \dots, v_n\}$, 则对应的模糊评级矩阵可以记为 $R = [m_i(A(v_j))]_{m \times n} (i=1,2,\dots,m; j=1,2,\dots,n)$ 。

3 Web软件可信性评估实例分析

针对已经建立的 Web 软件可信性评级指标体系,选取某企业拟立项的网上银行软件项目作为评估对象来说明上述算法的有效性和实用性。

设评价对象为 p , 将本文第 1 章提到的 Web 软件可信性评价指标体系记为 $C = \{C_1, C_2, C_3\} = \{\{c_{11}, c_{12}, c_{13}, c_{14}\}, \{c_{21}, c_{22}, \dots, c_{25}\}, \{c_{31}, c_{32}, c_{33}\}, \{c_{41}, c_{42}\}, \{c_{51}, c_{52}\}\}$, 令评语集为 $V = \{v_1: \text{非常不可信}; v_2: \text{有点不可信}; v_3: \text{一般可信}; v_4: \text{很可信}; v_5: \text{非常可信}\}$, 对应软件可信性级别为 $(T_1, T_2, T_3, T_4, T_5)$ 。

根据模糊评价的原理,按照如下的步骤来对网上银行软件项目 p 的可信性进行评价。

1) 确定指标的权重

根据德尔菲法进行专家访谈,得到评价指标的重要性排序表,再运用 2.1 节方法计算出评价指标体系中各级指标的权重,分别记为向量 w_1, w_2, w_3, w_4, w_5 。根据专家成员的选取应该满足鲜明的代表性、权威性和公正性等原则,邀请到 30 位熟悉网上银行软件的专家来确定指标权重,分为三组,分别为教授组、软件开发组 and 用户组,记为组 1、组 2 和组 3。各组专家在进行了相互独立的若干轮讨论之后,给出了各级可信属性重要性排序的判断表,形成各级指标的重要性排序矩阵,重要性排序矩阵及相关的计算结果如表 2 所示。

由计算过程的有效性和收敛性可以得到网上银行软件项目可信性评价指标体系一级指标集的权重向量为

$$w = (0.2961, 0.2223, 0.1782, 0.1820, 0.1214)$$

二级指标的权重向量为

$$w_1 = (0.3412, 0.2779, 0.2257, 0.1552)$$

$$w_2 = (0.1442, 0.1329, 0.1896, 0.2459, 0.2874)$$

$$w_3 = (0.2834, 0.3290, 0.3876)$$

$$w_4 = (0.5235, 0.4765)$$

$$w_5 = (0.5364, 0.4636)$$

2) 构造模糊评价矩阵

根据 2.2 节中介绍的方法,综合专家组对于各个二级指标的评价信息构造模糊评价矩阵,记为矩阵 R_1, R_2, R_3, R_4, R_5 。

继而从教授组、软件开发组 and 用户组三个组别中分别再邀请到三位更为熟悉网上银行软件项目的专家对软件可信性进一步进行评价。专家根据对拟采用的网上银行软件项目的相关资料进行分析之后,结合自己的知识、经验以及个人的偏好,给出各个二级指标在评价等级上的置信度,评价信息如表 3 所示。

表 2 重要性排序矩阵及相关的计算结果

指标	组 1	组 2	组 3	认知度	权重
C_1	1	1	2	0.966 1	0.296 1
C_2	2	4	1	0.837 1	0.222 3
C_3	3	5	3	0.644 7	0.178 2
C_4	4	2	4	0.708 1	0.182 0
C_5	5	3	5	0.515 8	0.121 5
C_{11}	1	2	2	0.574	0.341 2
C_{12}	3	1	3	0.828	0.277 9
C_{13}	2	3	4	0.65	0.225 7
C_{14}	4	4	1	0.478 2	0.155 2
C_{21}	5	3	4	0.762 4	0.144 2
C_{22}	4	4	5	0.887 9	0.132 9
C_{23}	3	5	3	0.765 9	0.189 6
C_{24}	1	2	2	0.873 1	0.245 9
C_{25}	2	1	1	0.893 2	0.287 4
C_{31}	3	2	3	0.695	0.283 4
C_{32}	2	3	1	0.768 9	0.329 0
C_{33}	1	1	2	0.953 9	0.387 6
C_{41}	1	2	1	0.875 4	0.523 5
C_{42}	2	1	2	0.798 6	0.476 5
C_{51}	2	1	1	0.875 4	0.536 4
C_{52}	1	2	2	0.917 3	0.464 6

表 3 关于网上银行软件的评价信息

指标	专家 1	专家 2	专家 3
C_{11}	$v_3(0.3)v_4(0.7)$	$v_3(0.6)v_4(0.5)$	$v_3(0.2)v_4(0.8)$
C_{12}	$v_3(0.2)v_4(0.7)v_5(0.1)$	$v_3(0.2)v_4(0.6)v_5(0.2)$	$v_3(0.3)v_4(0.6)v_5(0.1)$
C_{13}	$v_2(0.3)v_3(0.5)v_4(0.2)$	$v_2(0.2)v_3(0.6)v_4(0.2)$	$v_2(0.2)v_3(0.7)v_4(0.1)$
C_{14}	$v_3(0.2)v_4(0.7)v_5(0.1)$	$v_3(0.4)v_4(0.5)v_5(0.1)$	$v_3(0.3)v_4(0.5)v_5(0.2)$
C_{21}	$v_3(0.2)v_4(0.7)v_5(0.1)$	$v_3(0.4)v_4(0.5)v_5(0.1)$	$v_3(0.3)v_4(0.5)v_5(0.2)$
C_{22}	$v_2(0.3)v_3(0.5)v_4(0.2)$	$v_2(0.4)v_3(0.4)v_4(0.2)$	$v_2(0.4)v_3(0.3)v_4(0.3)$
C_{23}	$v_4(0.7)v_5(0.3)$	$v_4(0.6)v_5(0.4)$	$v_4(0.5)v_5(0.5)$
C_{24}	$v_4(0.5)v_5(0.5)$	$v_4(0.6)v_5(0.4)$	$v_4(0.3)v_5(0.7)$
C_{25}	$v_3(0.2)v_4(0.8)$	$v_3(0.1)v_4(0.9)$	$v_3(0.3)v_4(0.7)$
C_{31}	$v_1(0.4)v_2(0.5)v_3(0.1)$	$v_1(0.2)v_2(0.6)v_3(0.2)$	$v_1(0.4)v_2(0.1)v_3(0.5)$
C_{32}	$v_1(0.5)v_2(0.3)v_3(0.2)$	$v_1(0.4)v_2(0.3)v_3(0.3)$	$v_1(0.3)v_2(0.2)v_3(0.5)$
C_{33}	$v_4(0.7)v_5(0.3)$	$v_4(0.6)v_5(0.4)$	$v_4(0.5)v_5(0.5)$
C_{41}	$v_1(0.4)v_2(0.6)$	$v_1(0.3)v_2(0.7)$	$v_1(0.2)v_2(0.8)$
C_{42}	$v_1(0.1)v_2(0.9)$	$v_1(0.2)v_2(0.8)$	$v_1(0.5)v_2(0.5)$
C_{51}	$v_1(0.2)v_2(0.8)$	$v_1(0.3)v_2(0.7)$	$v_1(0.8)v_2(0.2)$
C_{52}	$v_1(0.4)v_2(0.6)$	$v_1(0.7)v_2(0.3)$	$v_1(0.3)v_2(0.7)$

通过原始评级信息,再结合相应的合成结果,可以验证改进的证据合成方法在综合专家评价信息时所具备的良好聚焦性;同时,观察 C_{14} 的评级信息及其对应的合成结果也可以发现,改进的合成方法当异常评价信息存在时仍然可以得到有效的合成结果。因此,采用上述改进的证据合成方法来处理专家对网上银行软件项目的可信性专家意见综合处理、构建模糊评价矩阵和得出评价结果的过程是具备可靠性、有效性和可识别性的。

3) 计算模糊评价向量

计算出一级可信指标的模糊综合评价向量 $B_i (i=1,2,3,4,5)$:

$$B_i = w_i R_i \quad (19)$$

从而,基于 B_i 构建出一级可信指标对象的模糊评价矩阵 $R =$

$[B_1^T, B_2^T, B_3^T, B_4^T, B_5^T]$, 计算出网上银行软件项目整体可信性的模糊综合评价向量 B 为

$$B = wR \quad (20)$$

根据式 (18) (19) 计算可得 $B_1 = (0.0599, 0.1325, 0.2699, 0.4804, 0.0572)$, $B_2 = (0, 0.1589, 0.2452, 0.4081, 0.1878)$, $B_3 = (0, 0.0759, 0.3107, 0.4101, 0.2029)$, $B_4 = (0, 0.3483, 0.4450, 0.2067, 0)$, $B_5 = (0, 0.3245, 0.3391, 0.3364, 0)$, $B = (0.1897, 0.2506, 0.1716, 0.1934, 0.1938)$ 。

4) 对模糊综合评价结果进行识别

由于本文中对软件可信性评价是有序划分的, 最大隶属度识别准则在此并不适用, 因此本文改用置信度识别准则。

设置信度为 λ ($0.5 < \lambda < 1$, 通常取 0.6 或 0.7), 本文取 $\lambda = 0.6$ 。

$$\text{若 } k_i = \min \{ k; \sum_{l=1}^k b_{il} \geq \lambda, 1 \leq l \leq k \} \quad (21)$$

则 x_i 属于第 k 个评价等级 C_k 。

按照评分准则进一步计算各一级指标可信属性得分:

$$q_{xl} = \sum_{i=1}^n n_i b_{il} \quad (22)$$

其中: n_l 表示等级 C_k 的分值, 这里取 $n = 8 - l$ 。由此, 可以根据 q_{xl} 的大小对各个一级指标的可信性进行排序。

从表 4 所示的分析结果可以看出, 该企业拟立项的网上银行软件的整体可信性等级为 T_3 (即一般可信), 在可以接受的范围之内。再结合各个一级可信属性的评价等级及得分, 对应可信性决策规则集 (表 5) 可以得出, 该网上银行软件项目需要进行总体修改后再开发, 并且, 在进一步的修改工作中要着重对可信性等级较差的效率和安全性进行重点关注。

表 4 网上银行软件可信性评价等级及得分

一级指标	可靠性	可用性	功能性	效率	安全性	软件可信性
评价等级	IV	IV	IV	III	III	T_3
得分	2.6	2.187 4	2.055 9	2.728 2	2.315 3	2.465 7

表 5 可信性决策规则集

可信性级别	可信性的评价集	可信性综合评价值的取值区间	所对应的决策
T_1	非常不可信	$[0, 0.2]$	大量修改后再评估
T_2	有点不可信	$(0.2, 0.4]$	小幅度修改后再评估
T_3	一般可信	$(0.4, 0.6]$	总体修改后开发
T_4	很可信	$(0.6, 0.8]$	局部优化后开发
T_5	非常可信	$(0.8, 1.0]$	无风险应用、开发

4 结束语

本文在已有的研究基础上, 应用因子分析法建立并完善了 Web 软件可信属性指标体系, 引入结构熵值法对各个可信属性进行权重的确定, 重要性排序提高了专家在进行主观赋权时的可操作性和简便性, 熵值法提高了客观赋权时的可靠性, 两者的结合对于 Web 软件可信属性权重的研究降低了工作的难度并提升了工作的效率。最后, 鉴于证据理论处理不确定性的有效性, 采用了改进的证据合成理论, 提高了对评价信息中不确定性问题处理的有效性, 用其构造出的模糊评价矩阵, 利用置信度评价准则对评价结果进行等级划分和得分计算, 增强了对评价结果的可识别性和可靠性。综合利用 SPSS 17.0 和

MATLAB 7.0 软件, 证实了上述方法在对 Web 软件可信性进行评估的实用性和有效性。本文提出的方法不仅适用于对 Web 软件进行可信性的评估, 也可以扩展到其他领域软件可信性的评估。

软件可信性的评估是一个复杂的综合过程, 下一步的工作中, 仍需要加强现代化方法如元模型、数据挖掘、形式化语义分析以及量化方法等关键领域的技术研究, 以更好地促进软件可信性评估方法的发展。在解决软件可信性评估问题时, 为了保证最终制定的改进策略更为切实有效, 在可信性决策规则集的建立上也要进一步研究决策专家的多视角问题, 不断细化和优化决策规则。

参考文献:

- [1] BASILI V, DONZELLI P, ASGAR S. A unified model of dependability: capturing dependability in context [J]. *IEEE Software*, 2004, 21(6): 19-25.
- [2] UZUNOV K, NGUYEN T. Dependability of software in airborne mission systems, DSTOTR-2111 [R]. [S. l.]: Deference Science and Technology Organization, Australian Government Department of Deference, 2008.
- [3] PAUL R, YEN I L, BASTANI F, et al. An ontology-based integrated assessment framework for high-assurance systems [C]//Proc of IEEE International Conference on Semantic Computing. Washington DC: IEEE Computer Society, 2008: 386-393.
- [4] WHITWORTH B, FJERMESTAD J, MAHINDA D. The Web of system performance [J]. *Communications of the ACM*, 2006, 49(5): 93-99.
- [5] WATER J G. Software dependability evaluation based on Markov usage models [J]. *Performance Evaluation*, 2000, 40(4): 199-222.
- [6] SHI Hui-ling, MA Jun, ZOU Feng-yi. Software dependability evaluation model based on fuzzy theory [C]//Proc of International Conference on Computer Science and Information Technology. 2008: 102-106.
- [7] MA Yan-fang, ZHANG Min. A computation model of trustworthy degree [C]//Proc of International Symposium on Intelligent Information Technology Application Workshops. 2008: 523-526.
- [8] MEI Lin, XU Yong-sen. An adaptive dependability model of component-based software [J]. *ACM SIGSOFT Software Engineering Notes*, 2003, 28(2): 10-14.
- [9] ZHANG Yang, FANG Bin, XU Chuan-yun. Trustworthy metrics models for internetware [J]. *Wuhan University Journal of Natural Sciences*, 2008, 13(5): 547-552.
- [10] ZHU Ming-xun, LUO Xin-xing, CHEN Xiao-hong, et al. A non-functional requirements tradeoff model in trustworthy software [J]. *Information Sciences*, 2012, 191: 61-75.
- [11] BEHKAMAL B, KAHANI M, AKBARI M K. Customizing ISO 9126 quality model for evaluation of B2B applications [J]. *Information and Software Technology*, 2009, 51(3): 599-609.
- [12] 李仕峰, 杨乃定, 刘效广. 基于熵和证据理论的 NPD 项目复杂性模糊评价 [J]. *管理工程学报*, 2013, 27(1): 121-126.
- [13] JOUSSELME A L, GRENIER D, BOSSE E. A new distance between two bodies of evidence [J]. *Information Fusion*, 2001, 2(2): 91-101.