

基于多属性熵权合成的软件可信等级评估方法

王 犇, 周兴社, 杨亚磊

(西北工业大学 计算机学院, 陕西 西安 710072)

摘 要: 提出了一种基于多属性熵权合成的软件可信等级评估方法, 该方法基于证据分类对多元可信证据进行合成, 再通过不确定熵计算各可信属性的权重值, 进而合成待评测软件的可信级别. 该方法着重解决了软件可信性评估中可信证据合成时证据冲突, 以及多属性权重分配等问题, 从而使软件可信评估的结果更加准确和真实. 最后通过具体实施验证了该方法的可行性.

关键词: 软件可信性评估; 可信属性; 证据合成; 不确定熵

中图分类号: TN402

文献标识码: A

文章编号: 1000-7180(2014)06-0021-04

An Entropy-weighted Multi-attribute Combination Method for Software Trustworthiness Classification Assessment

WANG Ben, ZHOU Xing-she, YANG Ya-lei

(School of Computer, Northwestern Polytechnical University, Xi'an 710072, China)

Abstract: We introduce an entropy-weight multi-attribute combination method for software trustworthiness classification assessment, which using evidence classification to combine multiple trust evidence, employing uncertain entropy to calculate the weight of each trust attributes, and then combining the trust level of the evaluated software. The method has solved the trust evidence conflict, as well as multiple attribute weights allocation problem, so that the result is more accurate. At last, the feasibility of this method is verified by specific implementation.

Key words: software trustworthiness assessment; trust attribute; evidence combination; entropy

1 引言

目前的可信软件构造和可信性度量与评测方法较为缺乏, 已有方法亦存在很多不完善的方面^[1]. 例如, 现有软件可信性评估方法缺乏对软件可信性度量的统一定义和准确说明, 对可信属性、可信证据等概念及其之间的关系缺乏确切定义, 没有充分考虑可信证据的多源性、不确定性等特征, 使得评估结果无法真实、准确地反映待评测软件的可信性.

本文提出了一种基于多属性熵权合成的软件可信等级评估方法. 该方法充分考虑了软件可信属性

的多元性, 以及可信证据的多样性和不确定性, 解决了可信证据之间的冲突, 以及证据本身的不确定性对软件可信性评估的影响, 其评估结果是 TRUST-IE-STC 规范^[2-4]所定义的软件可信等级, 从而使软件可信评估的结果更加准确和真实.

2 基于多属性熵权合成的软件可信等级评估方法

通过对可信评估现有技术的分析, 发现在利用 D-S 证据理论^[5-6]合成可信证据时, 没有考虑证据之间的冲突问题和证据本身的不确定性问题, 然而在

收稿日期: 2013-10-08; 修回日期: 2013-11-29

基金项目: 国家“八六三”计划项目(2009AA010308); 国家核高基项目(2009ZX01043)

称为可信证据的不确定熵. 某可信证据的不确定熵越大说明该可信证据包含的信息量越少, 该可信证据对合成结果的支持程度也就越小, 因此, 其对应的权重就应越小.

(1) m_i 的不确定熵定义如下:

$$E(m_i) = - \sum_j D_i(B_j) \ln D_i(B_j) \quad (8)$$

(2) 定义证据的确定度为

$$F(m_i) = a \cdot e^{-E(m_i)} \quad (9)$$

式中, a 为可选系数.

(3) 定义合成的证据 m^i 的加权系数为

$$d_l = \frac{\sum_{m_i \in M_l} F(m_i)}{\sum_{k=1}^n \left\{ \sum_{m_i \in M_l} F(m_i) \right\}} \quad l = 1, 2, \dots, n \quad (10)$$

对所有分类合成证据进行加权平均即为最终可信属性支持度:

$$H_j = \sum_i d_i m^i \quad (11)$$

式中, i 代表不同证据的分类, j 代表不同可信属性.

2.3 基于规则评估软件可信等级

对所有可信属性的支持度加权平均即为最终软件的可信度:

$$Tr = \sum_j \epsilon_j H_j \quad (12)$$

式中, ϵ_j 是专家赋予各个可信属性的权重, H_j 是通过权重计算得到的证据对可信属性的支持度.

参照 TRUSTIE-STC 规范定义软件可信等级评估规则:

If($0 \leq Tr < 0.2$) Then $Tr_Level = 1$;
 If($0.2 \leq Tr < 0.4$) Then $Tr_Level = 2$;
 If($0.4 \leq Tr < 0.6$) Then $Tr_Level = 3$;
 If($0.6 \leq Tr < 0.8$) Then $Tr_Level = 4$;
 If($0.8 \leq Tr \leq 1$) Then $Tr_Level = 5$;
 Tr_Level 代表软件的可信等级.

TRUSTIE-STC 规范中明确定义了软件可信等级的具体含义, 该规则在此基础上明确了软件可信性的度量值与软件可信等级之间的关系. 通过软件可信等级的评估为相关人员做出进一步的决策提供了可靠的参考.

采用基于分类及不确定熵的合成方法可以有效

的提高软件可信评估的可靠性和真实性. 传统的 D-S 证据合成规则采用的是乘性策略, 当可信证据间冲突较小时, 测度向不确定性小的属性集中, 而单一错误的可信证据可能导致合成结果不正确. 而基于可信证据分类及不确定熵的合成策略是乘性策略和加性策略的折衷, 首先, 通过将可信证据分类避免了存在较大冲突的可信证据间的合成, 接着, 将同类可信证据采用乘性的 D-S 证据合成方法进行合成, 保证了良好的收敛性; 最后采用加性策略二次合成不同类别的可信证据, 保证了良好的可靠性^[7].

3 示例

假设通过分析用户需求, 得到软件所关注的可信属性有: 可靠性(H_1)、安全性(H_2)、可生存性(H_3)、可用性(H_4). 同时利用专家咨询法确定软件可信属性之间的权重: ϵ_1 、 ϵ_2 、 ϵ_3 、 ϵ_4 ;

参照 TRUSTIE-STC 规范所定义的软件可信证据框架, 获得支持这些可信属性的证据项, 表 1 列举了部分可信证据.

表 1 可信证据

可信属性	可信证据
可用性	(1) 功能符合性 $X = cf * (1 - A/B)$ ($0 \leq X \leq 1$), A = 在评价中检测有问题的功能数, B = 被评价的功能数, cf = 评测机构的影响因子(信任度); (2) 功能准确性 $X = 1 - A$ ($0 \leq X \leq 1$), A = 软件产品没有提供所需精确度的功能数的比例; (3) 资源使用率: 内存使用大小, CPU 使用率等; (4) 用户关于软件功能的评价; (5) 软件投入使用的时间、使用频度以及发行量;
可靠性	(1) 需求变更数; (2) 平均失效时间 MTTF; (3) 平均失效间隔时间 MTBF; (4) 用户关于软件可靠性的评价;
可生存性	(1) +++ 抗攻击性 $X = \{Tr/Ta\}$ Ta = 被攻击次数, Tr = 成功抵抗攻击次数; (2) 恢复性 $X = \{Tr/Ta\}$ Ta = 遭受攻击次数, Tr = 成功恢复次数; (3) 用户关于软件可生存性的评价;
安全性	(1) MTTUF 安全关键软件发生第一次危险失效前的平均防患时间; (2) 用户关于软件安全性的评价.

对软件可信证据进行分类, 例如: H_i 证据经过度量产生 mass 函数有 m_1, m_2, m_3, m_4, m_5 共有 5 个.

根据证据分类算法对证据进行分类,分类结果如表 2 所示.

表 2 可信证据分类结果

	B_0	B_1	B_2	B_3	B_4	B_5
m_1	✓	✓	✓			
m_2		✓	✓	✓		
m_3		✓		✓		
m_4	✓	✓		✓	✓	
m_5		✓	✓	✓		
	T_{B_0}	T_{B_1}	T_{B_2}	T_{B_3}	T_{B_4}	T_{B_5}

说明:✓ 代表了证据支持某一基元.

从表 2 可以得知:

$$T_T = \{(m_1, m_4), (m_1, m_2, m_3, m_4, m_5), (m_1, m_2, m_5), (m_2, m_3, m_4, m_5), (m_4)\},$$

$$T_D = \{(m_1, m_4), (m_4)\}.$$

利用证据分类算法 H_i 证据分为两类: $(m_1, m_4), (m_2, m_3, m_5)$.

把上面的分类结果利用下面的 D-S 证据公式进行合成:

$$m(A) = \frac{1}{1 - k} \sum_{A_i \cap B_j \cap C_l \cap \dots = A} m_1(A_i) \cdot m_2(B_j) \cdot$$

$$m_3(C_l) \cdot \dots$$

其中

$$k = \sum_{A_i \cap B_j \cap C_l \cap \dots = \emptyset} m_1(A_i) \cdot m_2(B_j) \cdot m_3(C_l) \cdot \dots$$

利用不确定熵即式(8)~(10)确定各分类合成结果的权重 d_i ,利用下面公式合成各个可信属性的支持度:

$$H_j = \sum_i d_i m^i.$$

迭代公式(8)~(11),计算所有软件可信属性的支持度;最后,利用下面公式合成软件可信等级:

$$Tr = \sum_j \epsilon_j H_j.$$

式中, ϵ_j 是专家赋给可信属性的权重.

最终针对评估结果 Tr 的值,按照软件可信等级评估规则评定软件的可信等级.

4 结束语

本文遵循 TRUSTIE-STC 规范,提出了一种基于证据分类及不确定熵的软件可信等级评估方法,通过具体实施方法验证了基于多属性熵权合成的软件可信等级评估方法的可行性.

参考文献:

- [1] Li Meng, Zhou Xianzhong, Wang Jianjun, et al. A perspective of software trustworthiness based on distrustable factors [C] // Proc of IEEE International Conference on Networking, Sensing and Control. Japan: Okayama, 2009: 873-878.
- [2] “可信的国家软件资源共享与协同环境”课题组. TRUSTIE 软件可信分级规范 [V2. 0] [EB/OL]. (2009-05-30)[2013-06-15]. [http://www.trustie.net/UserFiles/File/\[TRUSTIE-STC\]软件可信分级规范\[V2.0\]-200906.pdf](http://www.trustie.net/UserFiles/File/[TRUSTIE-STC]软件可信分级规范[V2.0]-200906.pdf).
- [3] “可信的国家软件资源共享与协同环境”课题组. TRUSTIE 软件可信证据框架规范 [V2. 0]. (2009-05-30)[2013-06-15]. [http://www.trustie.net/UserFiles/File/\[TRUSTIE-STC\]软件可信证据框架规范\[V2.0\]-200906.pdf](http://www.trustie.net/UserFiles/File/[TRUSTIE-STC]软件可信证据框架规范[V2.0]-200906.pdf).
- [4] 郎波, 刘旭东, 王怀民, 等. 一种软件可信分级模型 [J]. 计算机科学与探索, 2010, 4(3): 231-239.
- [5] Dempster A P. Upper and lower probabilities induced by a multivalued mapping [J]. The Annals of Mathematical Statistics, 1967, 38(4): 325-339.
- [6] Shafer G A. Mathematical theory of evidence [D]. Princeton N J: Princeton University Press, 1976: 19-63.
- [7] 何兵. 基于分类及不确定熵的 DS 证据合成及判决方法 [J]. 北京航空航天大学学报, 2003, 29(10): 927-930.

作者简介:

王 犇 男, (1980-), 博士研究生. 研究方向为分布式计算与可信计算.