

基于模糊偏序关系的软件测试评价方法

姚奕, 刘晓明, 黄松

(解放军理工大学指挥自动化学院 南京 210042)

【摘要】讨论了模糊理论在软件测试质量评价方面的可行性;提出了软件测试质量评价指标体系的概念和软件测试质量评价方法,即基于模糊偏序关系的评价方法;对该方法的优缺点及适用条件进行了分析;通过实例计算验证了该方法在评价软件测试质量方面的有效性。

关键词 模糊偏序关系; 指标体系; 软件度量; 软件测试质量
中图分类号 TP311 文献标识码 A

An Assessment Method for Fuzzy Partial Ordering Relation-Based Software Testing Quality

YAO Yi, LIU Xiao-ming, HUANG Song

(Institute of Command Automation, PLA University of Science and Technology Nanjing 210042)

Abstract Software testing scheme has the important influence with reliability and efficiency of software testing. Testing scheme can be selected according to the assessment of software testing quality. This paper mainly discusses the feasibility of fuzzy theory on assessing software testing quality. The concept of software testing quality indicator system and a method for assessing software testing quality, namely assessment method based on fuzzy partial ordering relation are provided. The advantages and drawbacks of this method and its application conditions are analyzed. By the calculation results of actual examples, the availability of the method is verified on the aspect of assessing software testing quality.

Key words fuzzy partial ordering relation; indicator system; software metrics; software testing quality

测试在软件开发过程中备受关注,在传统的软件工程中有一个明确独立的测试阶段。随着软件危机的频频出现以及人们对于软件本质的进一步认识,测试的地位得到了很大的提高^[1]。随着软件测试理论的不完善,软件的评价和软件质量的度量获得了很大的进步。选择科学合理的评价方法成为软件测试质量评价的首要问题,也是当前软件测试质量评价研究中的难点。本文以软件测试方法为基础,提出了通过建立软件测试质量评价指标体系,运用模糊理论进行软件测试质量评价的方法;并结合解放军理工大学指挥自动化学院软件测评中心的实践经验,对软件测试质量进行定性与定量的综合评价^[2-3]。

1 运用模糊理论评价软件测试质量的可行性

1.1 软件测试质量指标体系

对测试项目的整体质量做一个客观评价时需有

一个度量标准来实施,因此应建立一个软件测试质量指标体系。本文以软件测评实验室软件测试项目的度量指标为基础构建评价指标体系。按照传统方法对软件测评实验室承担的软件测试项目,存在项目度量、过程度量、产品度量三个方面的度量。项目规模、工作量、成本、进度、资源等都属于项目度量的范畴,在过程度量中需关注生产率和有效性等方面内容;在产品度量中需关注产品规模、产品质量^[4]。建立软件测试质量指标体系所需的度量标准与上述的三个方面的度量标准不完全一致,存在交叉关系,因此软件测试质量指标体系实际上是在一些软件度量的基础上做了一定的选取和改进,其软件度量必须能反映出软件测试活动本身的质量。本文基于解放军理工大学指挥自动化学院软件测评中心的质量度量指标构建评价指标体系,其部分内容如表1所示。

表1 软件测试质量评价指标体系(部分)

序号	质量指标	赋值1	赋值0
1	所用时间	低于所有测试方案耗时的均值	高于均值
2	人员投入	低于所有测试方案用人的均值	高于均值
3	测试用例生产率	高于所有测试方案用例生产率均值	低于均值
4	测试用例的发现效率	高于所有测试方案用例发现效率均值	低于均值
5	测试用例的深度覆盖率	高于90%	低于90%
6	测试用例的广度覆盖率	高于90%	低于90%
7	测试发现问题的严重性	同已发现的缺陷相比,缺陷级别较高	级别较低
8	测试进度变化率	实际测试进度相对计划进度的偏差率低于8%	高于8%

1.2 运用模糊理论评价软件测试质量的可行性

通过软件测试质量指标体系可以看出,由于软件测试的特殊性,虽然指标值在承诺范围内,但仅用“好”与“不好”来描述测试的质量并不符合实际。测试方案之间的质量差异不很明显,质量的变化存在模糊性,因此在识别质量变化时,模糊方法比经典数学方法可行。运用模糊方法评价信息不是“非此即彼”,反映了测试人员主观评价测试质量的真实感受,可以更准确地描述测试人员对测试质量的感受程度,使软件测试质量的评价信息更接近实际。对模糊方法的研究也已经相对成熟,运用模糊方法对软件测试质量进行评价是可行的。

2 基于模糊偏序关系的软件测试质量评价方法

2.1 评价过程

2.2.1 软件测试质量指标的综合处理

设有 n 个被评价的测试方案,分别求出软件测试质量指标体系中 m 个定量指标在评价期内的均值,并对评价指标赋值。 a_{ij} 为第 i 个测试方案的第 j 个指标赋值($i=1, 2, \dots, n; j=1, 2, \dots, m$),可得出指标赋值矩阵为:

$$R = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{pmatrix}$$

随后确定各指标的权重为 $W=[w_1, w_2, \dots, w_m]^T$,其中, W 为权重向量; w_i 为第 i 个指标的权重,则有:

$$B=RW = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{pmatrix} \times \begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} \quad (1)$$

式中 B 为测试方案质量指标综合值向量; b_i 为第 i 个测试方案质量指标的综合值。由此可见,各指标权重的确定将对评价结果产生决定性的影响,因此如何正确合理地给出权值成为该方法中的一个难点。在实际运作中需要通过对测试方案涉及的数据资料进行整理,才能得到科学有效的权值。整理的具体方式如下:

1) 若获取的数据资料直接为对质量指标重要性的评价,如就某项质量指标收集数据资料时,测试人员可能对质量指标给出极其重要、很重要、重要、一般重要、不太重要、不重要等评价等级,这时一般利用每个质量指标评价等级的数目与本项质量指标的总评价数进行整理。假设对测试方案的第 i 个质量指标进行评价,参与评价的测试相关人员有 M 个,认为此项承诺达到评价等级 v_1, v_2, \dots, v_m 的人数分别为 M_1, M_2, \dots, M_m ,则 $a_{iv_1} = M_1/M, a_{iv_2} = M_2/M, \dots, a_{iv_m} = M_m/M$ 。即对所有的测试方案而言,其质量指标的评价($a_{iv_1}, a_{iv_2}, \dots, a_{iv_m}$)为已知。

2) 若获取的数据资料为绝对数或相对数,数据资料的整理过程如下:

(1) 将所获取的数据按某种标志分组,将不同性质的数据分开,相同性质的数据归在一起。每个质量指标一般按大小分组。分组界限包括组数、组距、组限,其中组数一般和评价等级数相同;组距为各评价等级中最大值与最小值之差,根据评价内容,可有等距与不等距两种;组限为评价等级的两个数据端点,每组的最小值为该组的下限,最大值为该组的上限,通常将达到上限的实际值归于数值高的一组。

(2) 将所有分组与各质量指标的评价等级相对应(分组时已考虑到),每组数据的个数即为对应的评价等级所拥有的数据个数。

(3) 计算各质量指标的权值。通过上述步骤已确定了各质量指标评价等级所拥有的数据个数,确定质量指标权值的方法与获取数据资料直接为所评价内容的质量等级时方法相同^[5]。

2.2.2 建立模糊偏序关系

(1) 给出二元相对比较级。二元相对比较级是根据对定性指标的经验判定和每个测试方案质量指标的综合值,由测试人员对每两个被评价测试方案

(x_i, x_j) X (X 为被评价测试方案的集合)给出数对 $(f_j(x_i), f_i(x_j))$, $0 < f_j(x_i) < 1$, $0 < f_i(x_j) < 1$ 。其意义是在 x_i 与 x_j 的比较中,若 x_i 具有某种特性的程度为 $f_j(x_i)$, 则 x_j 具有该特性的程度为 $f_i(x_j)$; 当 $i=j$ 时, 令 $f_i(x_j)=1$, 则可以得到模糊矩阵为:

$$R = \begin{pmatrix} 1 & f_2(x_1) & \cdots & f_n(x_1) \\ f_1(x_2) & 1 & \cdots & f_n(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ f_1(x_n) & f_2(x_n) & \cdots & 1 \end{pmatrix}$$

式中 R 为二元比较矩阵, R 具有模糊自反性^[6]。

(2) 求出 R 的传递闭包 \hat{R} , \hat{R} 具有模糊自反性, 故 $\hat{R} = R^{n-1}$ 。为减少计算量, 可以通过逐次平方法求得 \hat{R} , \hat{R} 是具有模糊自反性和模糊传递性的模糊关系, 因此, \hat{R} 是 X 上的模糊预序关系。

(3) 将模糊预序关系 \hat{R} 改造成 X 上的模糊偏序关系 Q , 其隶属度规定为:

$$Q(x_i, x_j) = \begin{cases} 1 & i = j \\ r_{ij} - r_{ji} & i \neq j, r_{ij} > r_{ji} \\ 0 & i \neq j, r_{ij} \leq r_{ji} \end{cases} \quad (2)$$

式中 r_{ij} 、 r_{ji} 为 \hat{R} 中的元素, Q 满足传递性在数学中已经得到证明^[7]。

2.2.3 确定测试质量优劣顺序

在建立模糊偏序关系这个关键性的步骤完成以后, 就可以利用模糊偏序关系矩阵对测试方案质量的优劣进行排序, 具体步骤如下: (1) 在模糊偏序关系矩阵 Q 中寻找第 i 列, 该列除了对角线上的元素是 1 外, 其余元素均为 0, x_i 为第一优越元, 排序时应排在第一位; (2) 去掉第一优越元所在的行和列, 得到新的 $n-1$ 阶模糊偏序矩阵, 用同样的方法得到第二优越元, 所对应的元素在排序时应排在第二位。依此类推, 按照求得优越元的先后次序对所有被评价的测试方案排出一定的优劣顺序。

2.2 本方法的特点及适用条件

从本文的评价过程可以看出, 基于模糊偏序关系的软件测试质量评价方法适用于同时评价多个测试方案, 从而得到相对优劣的排序。其优点是计算简便, 在对测试质量指标进行综合处理的基础上, 只需总体上给出测试方案在测试质量上的相对评价价值。然而该方法仍存在一定的缺陷, 在得到评价信息后, 需要采用主观方法处理评价信息, 进而建立模糊偏序关系, 对数据的处理难免会有误差^[8], 但可以采用增加测试人员的经验来进一步提高模糊偏

序关系准确性的方法。因此, 基于模糊偏序关系的评价方法是一种较为合理、有效的软件测试质量评价方法。

2.3 实例计算

本文假设有五个被评价的测试方案, 标记为方案1、方案2、方案3、方案4、方案5。针对这五个方案, 根据表1分别求出软件测试质量指标体系中定量指标在评价期内的实际值, 然后对评价指标进行赋值, 可得指标赋值矩阵为:

$$R = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix}$$

根据前面确定权重的方法, 计算出各质量指标权重向量为 $W=[0.14, 0.13, 0.14, 0.10, 0.12, 0.12, 0.13, 0.12]^T$, 进而由式(1)得到 $B=[0.73, 0.87, 0.63, 0.90, 0.61]^T$ 。根据软件测试质量指标综合值和对定性指标的判定建立模糊偏序关系, 可得到对五个测试方案优良程度的二元相对比较级的评价结果, 有:

$$R = \begin{pmatrix} 1.00 & 0.62 & 0.65 & 0.63 & 0.63 \\ 0.61 & 1.00 & 0.57 & 0.60 & 0.63 \\ 0.73 & 0.63 & 1.00 & 0.66 & 0.61 \\ 0.83 & 0.78 & 0.89 & 1.00 & 0.60 \\ 0.46 & 0.36 & 0.32 & 0.30 & 1.00 \end{pmatrix}$$

用逐次平方法可求出:

$$\hat{R} = R^4 = \begin{pmatrix} 1.00 & 0.65 & 0.66 & 0.66 & 0.64 \\ 0.64 & 1.00 & 0.64 & 0.64 & 0.64 \\ 0.71 & 0.68 & 1.00 & 0.68 & 0.68 \\ 0.85 & 0.78 & 0.88 & 1.00 & 0.63 \\ 0.46 & 0.45 & 0.45 & 0.45 & 1.00 \end{pmatrix}$$

则由式(2)可得:

$$Q = \begin{pmatrix} 1.00 & 0.01 & 0 & 0 & 0.18 \\ 0 & 1.00 & 0 & 0 & 0.19 \\ 0.05 & 0.04 & 1.00 & 0 & 0.23 \\ 0.19 & 0.14 & 0.20 & 1.00 & 0.18 \\ 0 & 0 & 0 & 0 & 1.00 \end{pmatrix}$$

按照上述排序方法, 由 Q 最终确定得到五个测试方案质量优劣的顺序为: 方案4、方案3、方案1、方案2、方案5。

(下转第509页)

Web服务发现性能的指标。查全率是指查询返回符合查询条件的Web服务与查询返回Web服务总数量的比率,查准率是指查询返回符合查询条件的Web服务与测试样本集中符合查询条件的Web服务的比率。分别用本文提出的算法和基于关键字的匹配的Web服务发现方法UDDI进行测试,得到的平均查全率为89%和56%,平均查准率为76%和43%,如图5所示。试验结果表明,本文提出的服务发现方法在查准率和查全率方面都要优于UDDI。

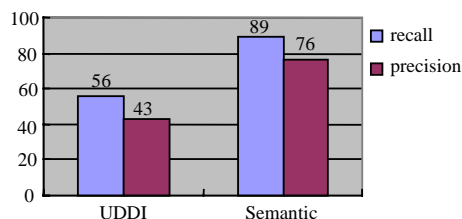


图5 性能测试查准率和召回率比较图

4 结束语

本文介绍了Web服务的本体语义描述和基于语义相似度的服务发现匹配算法。传统的基于WSDL的服务描述语言缺乏对服务的语义描述,本文用语义本体来描述服务以解决这个问题,基于本体的语义匹配算法可以实现传统的字符串匹配算法不具备的精确性和灵活性。本文提出的算法在查准率和查

全率上要优于现有UDDI上的基于关键字的服务发现机制。下一步研究还将提供更加详细的服务匹配信息以提高服务匹配的精确度以及研究在语义匹配的基础上进行服务的自动组合和执行等。

参 考 文 献

- [1] HAAS H. Web services activity statement[EB/OL]. <http://www.w3.org/2002/ws/Activity>, 2002-04-12.
- [2] UDDI org. Universal description, discovery, and integration of businesses for the web[EB/OL]. <http://www.uddi.org/>, 2004-10-01.
- [3] LEE T B, HENDLER J, LASSILA O. The semantic Web[J]. Scientific American, 2001, 284(5): 34-43.
- [4] PAOLUCCI M, KAWAMURA T, Payne T R, et al. Semantic matching of Web services capabilities[C]//In: Proceedings of the 1st International Semantic Web Conference (ISWC). Sardinia, Italia: [s. n.], 2002: 34-43.
- [5] OWL Services Coalition. OWL-S: Semantic markup for Web services[EB/OL]. <http://www.daml.org/services/owl-s/1.0/>, 2003-03-15.
- [6] DAVES A M. The optimal assignment problem[EB/OL]. <http://www.math.uwo.ca/~mdawes/courses/344/kuhn-munkres.pdf>. 2004-07-15.
- [7] LABC H P. Semantic Web programme Jena 2 ontology API[EB/OL]. <http://jena.sourceforge.net/ontology/index.html>. 2004-09-30.

编辑 孙晓丹

(上接第505页)

3 结 论

基于模糊偏序关系的软件测试质量评价是运用模糊理论定量地说明测试方案的质量,使测试质量的评价结果具有可比性。该方法简单易行,考虑问题具有系统性,数据的获得合理科学;采用客观方法处理评价信息,减少了评价结果的误差,对测试质量的评价更有说服力。在具体应用时,可根据所具备的条件和对评价结果的要求灵活采用这种方法。

参 考 文 献

- [1] PAUL C J. Software testing: a craftsman's approach, [M]. 第2版. 北京: 机械工业出版社, 2003.

- [2] 王 凌, 冯 惠, 杨根兴, 等. GJB 2434A-2004, 军用软件产品评价[S]. 北京: 总装备部军标出版发行部, 2004.
- [3] 冯 惠, 王 凌, 陈森方, 等. GJB 5236-2004, 军用软件质量度量[S]. 北京: 总装备部军标出版发行部, 2004.
- [4] MORMAN E F, SHARI L P. Software metrics: a rigorous and practical approach, [M]. 第2版. 北京: 机械工业出版社, 2004.
- [5] 杜吉泽. 市场分析[M]. 北京: 经济科学出版社, 2001.
- [6] 王培庄, 韩立岩. 应用模糊数学[M]. 北京: 北京经济出版社, 1989.
- [7] 贺仲雄, 王 伟. 决策科学从最优到满意[M]. 重庆: 重庆出版社, 1988.
- [8] 黄 沛. 模糊偏序关系及其在消费者广告偏好方面的运用[J]. 中国管理科学, 1997, 5(1): 57-64.

编辑 黄 莘