

基于模糊集理论的软件质量评估研究

陆 鑫, 廖建明

(电子科技大学计算机科学与工程学院 成都 610054)

【摘要】针对软件质量评估数据的模糊性问题,提出了基于模糊集理论实施评估的处理方案,改善传统评估方法中不精确性和复杂性的弊端,达到科学、客观、合理评估的目的;描述了模糊集理论解决现实世界模糊性问题的基本原理,并给出应用模糊逻辑与推理的多因子软件质量评估模型设计。针对模型的应用方法给出了一个软件质量评估实例,便于对多因子软件质量评估模型有更深入的理解。

关键词 评估模型; 模糊集理论; 多因子评价; 软件质量
中图分类号 TP182 文献标识码 A

Study of Software Quality Evaluation Based on Fuzzy Sets Theory

LU Xin, LIAO Jian-ming

(School of Computer Science and Engineering, University of Electronic Science and Technology of China Chengdu 610054)

Abstract This paper presents a scheme for evaluating software quality using fuzzy sets theory. The objective of the work is make the evaluation of software quality more scientific and reasonable than traditional methods. The fundament based on fuzzy sets theory for resolving fuzzy characters in realistic world is presented. The design of multi-factorial evaluation model for software quality is also provided. An example is given to demonstrate how to evaluate software quality.

Key words evaluation model; fuzzy sets theory; multi-factorial evaluation; software quality

与许多行业产品一样,质量对软件产品或软件系统都是十分重要的,它决定了产品或系统的生命力与市场。软件质量评估不但为软件质量的控制与保证提供可靠支持,也为软件的定型、验收、鉴定、评价和市场推广提供客观和科学的依据。在软件工程领域,不少人对软件质量的评估进行探索与实践,提出了有关软件质量的一些定义、度量方法、评价体系与标准。文献[1]提出了质量因子评估模型;文献[2-3]开发和发布了FURPS评估模型及ISO 9126/14598软件质量模型标准。然而,由于软件特性和质量因子等大都具有模糊特性,很多因子不能用确定的数值来表示,只能进行主观测度,难以直接用经典数学方法进行精确度量。另外,软件质量的特性还受多个因子影响,需要由多个质量因子进行综合评估。目前对软件质量的评估大多仍局限在定性分析上,很难真实反映软件质量的评估结果。建立一套科学、合理的软件质量定量评估模型显得尤为重要。

模糊集是表示和处理不确定性数据的重要理论方法^[4],它不仅处理不完全数据、噪声或不精

确数据,还可以用于开发数据的不确定性模型。本文将探讨如何运用模糊集理论实现软件质量的有效与合理评估。

1 软件质量评估方案设计

对软件质量进行评估,需要完成软件质量评估指标体系建立、质量因子度量方法确定、评价标准定义、评测数据综合评价处理等一系列工作。本文以ISO 9126标准定义的质量模型^[2]为基础,建立的软件质量评估指标体系如图1所示。依据软件质量层次特性模型,本文的方案建立了二级软件评估指标体系。该软件评估指标体系的第一级指标评价软件总体质量,包括功能性、可靠性、易用性、维护性、可移植性和效率六个特性指标;第二级指标评价软件质量的其他特性,如容错性、成熟性、易恢复性、可靠依从性等子特性。每个二级指标项又由多个质量因子(即度元)评价其子特性。可以看出,对软件质量的度量,是通过建立软件产品固有的质量指标评估体系,逐步细化其特性评价因子,实现对软件质量各层指标的度量。

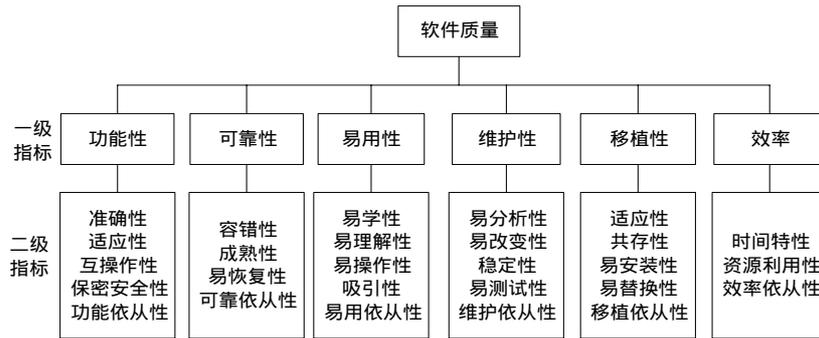


图1 软件质量评估体系

在软件质量定量评估方案中,需要对子特性的各质量因子进行度量,产生出软件质量评估的初始数据。文献[3]的模型给出了质量因子的一些度量方法。在本文的评估方案中,可以由评估专家基于评价标准定义进行模糊量化处理,得到每个质量因子的度量值,为综合评估的数据处理提供基本数据。

基于指标体系的软件质量综合评价过程的步骤如下:(1)对二级指标项的各质量因子,按其特定度量公式计算因子度量值,或由专家按评价标准给出因子度量值。(2)收集与整理质量因子度量值,按特定的评估模型计算各二级指标的评价值。(3)收集与整理二级指标的评价值,按特定的评价模型计算各一级指标的评价值。(4)收集与整理一级指标的评价值,按特定的评价模型计算软件质量的总评价值。

在软件质量评估中,不同指标和因子的重要性体现为其权重值的大小。值得注意的是,分配给各指标和因子的权值取决于特定软件产品和用户的需求。软件质量评估的结论可分为优秀、良好、合格和不合格四种。

需说明的是,在上述评估方案中,不少指标与因子具有模糊性,直接的数据处理难以实现定量评估的精确性。因此,在具有模糊特性的软件质量评估中需考虑如何合理、科学地进行数据评估处理。

2 模糊集理论

模糊集理论对模糊集定义了一整套与古典集合理论类似的运算规则及其推理逻辑。但模糊集理论突破传统经典集合的概念,引入隶属函数描述每个元素在集合中的归宿程度,并扩展集合的逻辑运算实现对模糊逻辑推理,以解决模糊特性问题。模糊集的定义如下^[5]:设 X 是一个对象空间, x 是 X 中的一个元素,那么 X 中的模糊集 A 可以定义为一组有序对,即 $A=\{(x,\mu_A(x))|x\in X\}$,其中 $\mu_A(x)$ 称作模糊集 A 的隶属函数,它将 X 中的每个元素映射到0到1之间的隶属值。隶属值反映元素 x 属于集合 A 的程度,也就是

模糊集描述模糊性的数学表示。从定义可以看出,模糊集实际上是古典集合的扩展,即当模糊集的隶属函数只有0和1这两个值时,它就退化为古典集合。

现对本文的评估模型将使用的模糊集运算规则进行说明如下:

(1)模糊集并运算规则。设 X 域有模糊集 $A=\{(x,\mu_A(x))|x\in X\}$ 和 $B=\{(x,\mu_B(x))|x\in X\}$,它们的并运算得到模糊集 C ,即 $C=A\cup B$, $\mu_C(x)=\mu_A(x)\vee\mu_B(x)=\max(\mu_A(x),\mu_B(x))$, $\forall x\in X$,其中 $\mu_C(x)$ 是模糊集 C 的隶属函数。

(2)模糊集交运算规则。设 X 域有模糊集 $A=\{(x,\mu_A(x))|x\in X\}$ 和 $B=\{(x,\mu_B(x))|x\in X\}$,它们的交运算得到模糊集 C ,即 $C=A\cap B$, $\mu_C(x)=\mu_A(x)\wedge\mu_B(x)=\min(\mu_A(x),\mu_B(x))$, $\forall x\in X$,其中 $\mu_C(x)$ 是模糊集 C 的隶属函数。

(3)max-min合成运算规则。设 R_1 和 R_2 分别是定义在 $X\times Y$ 和 $Y\times Z$ 上的两个模糊关系,它们的max-min合成运算结果为 $X\times Z$ 上的模糊集 $R_1\circ R_2=\{(x,z),\max\text{-min}(\mu_{R_1}(x,y),\mu_{R_2}(y,z))\}|\forall x\in X,\forall y\in Y,\forall z\in Z\}$ 或等价写为 $R_1\circ R_2=\vee[\mu_{R_1}(x,y)\wedge\mu_{R_2}(y,z)]$,其中 \vee 和 \wedge 分别代表max和min运算。当 R_1 和 R_2 用关系矩阵表示时, $R_1\circ R_2$ 运算可类似矩阵乘法运算,只是需将 \vee 和 \wedge 运算分别替代+和 \times 运算。

3 多因子软件质量评估模型

很难用经典数学模型精确评估具有模糊性质的质量因子,但可以把它们先转化为模糊集问题,然后再运用模糊集理论来解决此类问题,即将问题域的定性评估转化为模糊集的定量评估。在本文的评估方案中,每个指标项的评价都是由多因子确定的。因此,建立的软件质量评估模型将是基于多因子的,其评估模型定义如下:

设 $U=\{u_1, u_2, \dots, u_n\}$ 是一组用来进行评价的对象, u_i 表示第 i 个评估对象; $F=\{f_1, f_2, \dots, f_m\}$ 是评价过程中用到的一组基本因子,不同指标使用的评价因

子及其因子数量 m 是不一样的,它们由评估方案给出, $E=\{e_1, e_2, \dots, e_p\}$ 是用于评判因子的等级标准。本文模型中 $p=4$,并定义 $e_1=A$ 等级(优秀), $e_2=B$ 等级(良好), $e_3=C$ 等级(合格), $e_4=D$ 等级(不合格)。对于每个对象 $u \in U$ 都会有一个大小为 $m \times p$ 的评价矩阵 $R(u)$,该矩阵可看作模糊关系 $F \times E$ 的一个二维隶属函数。另外,每个对象 $u \in U$ 都对应一个与评价因子相关的权向量 $W(u)$,它的分量值为该指标各因子的参考权重。对象 u 的评估结果向量 $D(u)$ 是由权向量 $W(u)$ 与评价矩阵 $R(u)$ 进行max-min合成运算得到,如图2所示。

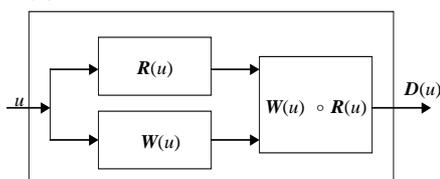


图2 多因子评估模型

该模型运算得到的评估结果向量以模糊集 $D(u)=[d_1, d_2, d_3, d_4]$ 的形式给出。然后选取 d_i 中最大值对应的等级作为该对象的评估结论,其中 $i=1 \sim 4$ 。例如,若 d_1 为最大值,则该对象的评估结论为A等级。如果同时存在两个或两个以上相同的最大值,如 d_1 、 d_2 均为最大值,则需进一步区分对象是偏重等级A还是偏重等级B。其方法是,将权向量 $W(u)$ 与评价矩阵 $R(u)$ 按标准矩阵乘运算,其结果向量中的最大分量值所对应的等级就是对象 u 的评估结论。

按照上述模糊集运算处理,可计算得到各指标的评估结果值,即A、B、C和D中的四个等级之一。最后再利用该模型对各指标的评估值进行类似处理,就可计算得到软件质量评估的总体结论,即优秀、良好、合格和不合格中的四种结论之一。

$$D=W \circ R=[0.40 \quad 0.10 \quad 0.20 \quad 0.20 \quad 0.10] \circ \begin{bmatrix} 0.30 & 0.60 & 0.10 & 0 \\ 0.50 & 0.30 & 0.10 & 0.10 \\ 0.40 & 0.45 & 0.10 & 0.05 \\ 0.20 & 0.60 & 0.10 & 0.10 \\ 0.60 & 0.20 & 0.20 & 0 \end{bmatrix} = [0.35 \quad 0.50 \quad 0.11 \quad 0.04]$$

运算得到的评估结果在向量 D 中,其中 d_2 分量值为最大值,可以确定“功能性”指标的评估值是偏重等级B,即该指标的评估结果为良好,其隶属值为50%。这个结果与专家们定性评估的合理结果是一致的。但本文模型得到的结果更具有数据说服力和模糊集理论的支持。

当使用本文模型进行软件质量总体评估时,需将一级指标作为评价因子,构造出软件质量的 R 评价

4 应用实例

下面以一个实例来简要说明多因子软件质量评估模型的具体应用方法。假设将对软件质量的“功能性”指标进行评估,得出其评估结果。使用多因子软件质量评估模型进行评估的过程如下:

(1) 确定“功能性”指标的基本因子为 $F=(f_1, f_2, f_3, f_4, f_5)=($ “准确性”,“适应性”,“互操作性”,“保密安全性”,“功能依从性”)。

(2) 确定基本因子的评判等级为: $E=(e_1, e_2, e_3, e_4)=(A, B, C, D)$

假设低层已经按多因子软件质量评估模型对“功能性”指标的各二级指标进行了评估,并得到各二级指标在评判等级上的隶属值,将它们构成一个在模糊关系 $F \times E$ 上的二维隶属评价矩阵 R 。假设各二级指标的 R 评价矩阵值为:

$$R = \begin{bmatrix} 0.30 & 0.60 & 0.10 & 0 \\ 0.50 & 0.30 & 0.10 & 0.10 \\ 0.40 & 0.45 & 0.10 & 0.05 \\ 0.20 & 0.60 & 0.10 & 0.10 \\ 0.60 & 0.20 & 0.20 & 0 \end{bmatrix}$$

该评价矩阵的每一行的值分别代表评估模型对“功能性”指标中各二级指标评判等级的隶属值。例如矩阵第三行的值表示互操作性指标隶属A级的值为40%、隶属B级的值为45%、隶属C级的值为10%、隶属D级的值为5%。

(3) 评估方案提供“功能性”指标的各因子权重向量为 $W=[0.40 \quad 0.10 \quad 0.20 \quad 0.20 \quad 0.10]$ 。

(4) 由权向量 W 与评价矩阵 R 进行max-min合成运算得到评估结果向量 D :

矩阵,其方法与上面类似。然后按模糊推理过程进行运算处理,就此可求出软件质量的总体评估结论。

5 结束语

本文提出了应用模糊集理论实施软件质量评估操作方案,其基本思想是基于模糊逻辑和模糊推理方法建立多因子软件质量评估模型,将原来具有模糊性的定性评估转化为基于模糊集理论的定量评

估,从而使得评估结果更具科学性与合理性。同时需指出的是,模糊集理论不仅可用来解决具有模糊性质的现实问题,也可应用于解决某些本身具有精确性质的现实问题。

参 考 文 献

- [1] CAVANO J P, McCall J A. A framework for the measurement of software quality[C]//Proc. ACM Software Quality Assurance Workshop NewYork: ACM, 1987, 133-139.
- [2] ISO/ IEC JTC1/ SC7/ WG6, ISO/ IEC 9126- 1: Information technology-software quality characteristics and metrics-Part 1: Quality model[S]. 2001.

- [3] ISO/ IEC JTC1/ SC7/ WG6, ISO/ IEC 14598 Part1-Part6: 2001. Information technology-evaluation of software product[S]. 2001.
- [4] PEDRYCZ W. Fuzzy set technology in knowledge discovery[J]. Fuzzy sets syst., 2001, 98: 279-290.
- [5] KLIR G J, YUAN B. Fuzzy sets and fuzzy logic: Theory and application[M]. Upper Saddle River: Prentice Hall, 1995.
- [6] LI H X, YEN V C. Fuzzy sets and fuzzy decision-making[M]. Boca Raton: CRC Press, 1995.
- [7] HAN J, KAMBER M. Data mining: Concepts and techniques [M]. America: Morgan Kaufmann Publishers, Inc., 2001.

编 辑 熊思亮

(上接第616页)

(7) 定制排队。定制排队是为允许具有不同最低带宽和延迟要求的应用程序共享网络而设计的。定制排队为不同协议分配不同的队列空间,并以循环方式处理队列,当特定协议的数据流被分配了较大的队列空间,也就获得了较优先的服务。定制排队比优先级队列更为公平,可以保证每一个特定的通信类型得到固定的可用带宽,同时在链路紧张的情况下,避免数据流企图超出预分配量限制的可能。对照评价标准可知,该方法改进后可用于防护DDoS攻击,在资源分配和使用时为不同业务提供优先级加权,改进思路与WFQ和WRED类似。

以上简略分析了当前的一些主流IP拥塞控制算法,并对其防护DDoS攻击的可行性进行了评价。可以看出,这些算法的防护能力存在着较大差异,其中的任何一种都需要改进,才能有效地应用于控制DDoS攻击所造成的网络拥塞。

4 结 束 语

针对防护DDoS攻击所进行的研究工作,应从网络拥塞和主机资源耗尽两个方面来着手开展,本文对前一个问题进行了系统的研究。在分析传统IP拥塞控制算法思想的同时,对其应用于DDoS攻击防护

的能力进行了评价,为进一步研究DDoS防护提供了依据和基础。

参 考 文 献

- [1] MIRKOVIC P R J, PRIER G. Attacking ddos at the source[C]// In Proc.: 10th IEEE international conference on network protocols. Paris: [s. n.], 2002.
- [2] VICISANO L, RIZZO L, Crowcroft J. TCP-like congestion control for layered multicast data transfer[C]//In Proc.: IEEE INFOCOM'98. San Francisco: [s.n.], 1998.
- [3] 罗万明, 林 闯, 阎保平. TCP/IP拥塞控制研究[J]. 计算机学报, 2001, 24(1): 1-18.
- [4] NAGEL J. On packet switches with infinite storage[J]. IEEE Trans. on Commun., 1987, 35: 435-438.
- [5] PAREKH A K, GALLAGER R G. A generalized processor sharing approach to flow control in integrated services networks: The single-node[J]. IEEE/ACM Trans. Networking, 1993, 1(3): 344-357.
- [6] FLOYD S. TCP and explicit congestion notification[J]. ACM Computer Communication Review, 1994, 24(5): 8-23.
- [7] ATHURALIYA S, LOW S, LAPSLEY D. Random early marking. in proceedings of the first International workshop on quality of future internet services[C]//QoS'2000.Berlin: [s.n.], 2000.
- [8] DEMERS A, KESHAV S, SHENKER S. Analysis and simulation of a fair queueing algorithm[C]//Proc. ACM SIGCOMM'89. Austin: ACM Press, 1989.

编 辑 熊思亮