计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2017,26(8):152−156 [doi: 10.15888/j.cnki.csa.005900]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

缺陷数据的相似性度量方法改进①

万　琳[1], 杨腾翔[1], 刘海宁[2]

[1](装甲兵工程学院 信息工程系, 北京 100072)
[2](中国兵器科学研究院, 北京 100089)

摘　要: 模糊聚类分析主要研究样本的分类问题. 本文利用模糊聚类方法对软件缺陷进行分类, 引入缺陷数据属性权重计算方法, 依据数据挖掘中的属性邻近性度量方法, 对缺陷数据进行相似度分析. 并按照属性类别进行分析, 不仅体现了缺陷数据属性间的形贴近程度, 而且体现了属性之间的距离贴近程度. 本文方法对软件缺陷数据进行分析并对比度量结果, 实验结果充分说明改进后的模糊聚类相似性度量方法在分类准确性方面有一定程度的提高.

关键词: 模糊聚类; 数据挖掘; 软件缺陷; 相似度; 属性权重

引用格式: 万琳,杨腾翔,刘海宁.缺陷数据的相似性度量方法改进.计算机系统应用,2017,26(8):152–156. http://www.c-s-a.org.cn/1003-3254/5900.html

Improvement of Similarity Measurement Method for Defect Data

WAN Lin[1], YANG Teng-Xiang[1], LIU Hai-Ning[2]

[1](Information Engineering Department, Academy of Armored Forces Engineering, Beijing 100072, China)
[2](China Academy of Ordnance Science, Beijing 100089, China)

Abstract: The study of fuzzy cluster analysis is mainly the classification of samples. In this paper, the fuzzy clustering method is used to classify the defects of software, and the method of attribute weight calculation is introduced. The similarity of defect data is analyzed with the method of attribute proximity in data mining. According to the category of attributes, it does not only reflect the degree of similarity between the attributes of the defect data, but also reflects the distance between the attributes. In this paper, the software defect data are analyzed and compared with the measurement results. The experimental results show that the improved fuzzy clustering similarity measurement method has somehow improved in classification accuracy.

Key words: fuzzy clustering; data mining; software defects; similarity; attribute weight

聚类是根据一定的规则, 按照数据的特征属性, 合理地划分给定未分类数据的集合, 得到数据分类的相关过程. 目前聚类分析方法主要有四种[1,2]: 基于层次的聚类方法, 基于划分的聚类方法, 基于图论的聚类方法, 基于密度和网格的方法等. 常用的聚类分析方法是 K-Means 方法, 该方法简单高效, 且伸缩性较好, 易于理解, 但是对于初始点的选择和 K 值的确定比较敏感. 在 VSM 中, 文本相似度计算方法主要有以下三种: Cosine, Dice, Jaccard, 其中应用最广泛的是 Cosine 法, 该方法简单高效, 适用于数值取值范围差异较小的数据集, 但是对于事物属性未加区分, 抽象化之后基于距离计算对象的相关性, 会导致计算结果不准确.

通常情况下, 事物属性包含四种: 标称属性, 序数属性, 二元属性, 数值属性. 传统聚类分析方法中单一的相似程度度量和距离程度度量不能完全描述事物特征[3]. 此外, 传统聚类方法没有考虑事物之间各属性的重要程度差异. 基于上述聚类分析和文本相似度计算方法的不足, 本文引入了相似程度和距离程度相结合

① 收稿时间: 2016-11-28; 采用时间: 2017-01-04

的方法, 分别计算缺陷数据标称属性和序数属性的相似度, 另外引入缺陷数据各属性的权值计算方法, 把属性对于分类规则的不同影响考虑在相似度的计算方法中, 提出一种既能考虑到缺陷之间属性的值贴近程度, 又考虑到缺陷之间属性的形贴近程度的计算方法, 最后综合到相似度度量中, 由此得到缺陷数据的相似度矩阵.

缺陷数据之间的相似性度量对于缺陷定位工作具有重要意义, 通过分析缺陷库与代码提交日志, 可以提取缺陷修复所对应的源文件位置关系. 通过这种链接关系可以建立已修复缺陷与相关源文件的位置关系, 通过本文的相似性度量方法建立待定位缺陷和已修复缺陷的关联关系, 利用这两层之间的关联关系定位待修复缺陷的位置. 同时, 该相似性度量方法对于缺陷数据的相关性分析也有一定的指导作用, 可用于预测同类缺陷的潜在属性和在软件中出现的概率大小. 总的来说, 本文结合模糊聚类和数据挖掘的分析方法对软件缺陷数据进行研究具有重要的现实意义.

1 缺陷数据的属性分类

软件缺陷是由许多特征属性表征的, 主要可分为标称属性和序数属性两大类. 例如表 1 中的缺陷数据, 其中 Test-1, Test-2 是标称属性, Test-3 是序数属性.

表 1 包含混合类型属性的缺陷数据表

缺陷编号	Test-1缺陷类型 (标称属性)	Test-2测试类型 (标称属性)	Test-3严重程度 (序数属性)	…
1	程序问题	文档审查	严重	…
2	设计问题	边界测试	重要	…
3	文档问题	功能测试	中等	…
4	程序问题	接口测试	次要	…

属性的类别不同, 则相似性的度量方法也不同. 为了提高缺陷数据相似性度量的准确度, 需要按照缺陷属性的不同类别分别进行分析, 依据单个属性计算缺陷数据之间的相似性, 最后将各个属性的相似性综合到缺陷的相似性度量中, 具体分析方法在下文进行讨论.

1.1 标称属性之间的相似性

数据挖掘中定义, 数据之间的邻近性包括相似性和相异性, 当对象 i 和 j 的标称属性相匹配时, 相似性 $\mathrm{sim}(i, j)=1$, 相异性 $\mathrm{d}(i, j)=0$. 当对象 i 和 j 不匹配时, 则相反.

由此可以看出, 对于 Test-1 属性, 除了缺陷 1 和缺陷 4 之外, 所有缺陷之间互不相似.

1.2 序数属性之间的相似性

标称属性一般采用布尔度量的方式, 若两个对象相匹配, 则相似度表示为 1, 若两个对象不匹配则表示为 0. 序数属性和标称属性的不同点在于抽象化的序数属性数值能够采用距离度量的方式计算缺陷的相似性, 并非只有“0”和“1”的度量方式, 所以序数属性之间的相似性度量一般包括以下两个步骤:

(1) 第 i 个对象的序数属性 f 值为 r_{if}, 属性 f 有 M_f 个有序的状态, 表示排位 1, …, M_f.

(2) 由于各个序数属性可能有大小差异较大的状态数表示, 为了使属性之间的相异性满足(0, 1)的范围内, 需要将属性值规范化. 用 z_{if} 表示第 i 个对象的 r_{if}, 公式如下:

$$z_{if}=\frac{r_{if}-1}{M_f-1} \tag{1}$$

(3) 然后将缺陷序数属性值进行距离度量, 对于单个属性而言, 利用欧几里得距离表示两个缺陷样本之间的相异性:

$$d(i,j)=\sqrt{(z_{if}-z_{jf})^2} \tag{2}$$

2 缺陷特征属性权重的计算

2.1 权重计算理论依据

根据缺陷数据属性的不同类别确定相似性后, 则要考虑不同属性对于缺陷之间综合相似度的影响. 事实上, 缺陷数据分类过程中各个属性的影响是不同的, 如果给各个属性分配相同权重, 则会导致分类结果不准确.

本文依据粗糙集理论和模糊聚类的方法[4-6]分析已知缺陷数据, 客观分配缺陷属性的权值. 首先将缺陷数据进行分类, 然后去除单个属性后再次分类计算该属性的重要性, 依次确定缺陷各个属性的权值. 具体步骤如下:

(1) 建立缺陷数据编码表, 将缺陷数据抽象化, 各个缺陷可表示为由 n 个抽象化特征值表示的向量, 则缺陷 $x_i=\{x_{i1}, x_{i2}, \ldots, x_{in}\}$, 最后所有的缺陷数据可以映射为特征空间的一个多维矩阵, 其数学表示方式如下:

$$X=\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \tag{3}$$

(2) 将矩阵(3)按照(1)式进行规范化后处理然后利用最大最小值的方法建立模糊相似矩阵 T, 计算公式如下:

$$x_{ij}=\sum_{k=1}^{n}(x_{ik}\wedge x_{jk})/\sum_{k=1}^{n}(x_{ik}\vee x_{jk}) \tag{4}$$

其中, i=1, 2, …, m; j=1, 2, …, n; m 为矩阵的行数, n 为矩阵的列数.

最后通过平方法将模糊相似矩阵不断自乘, 直到满足 $T^{2k}=T^{k}=T$, 建立具有对称性, 传递性, 自反性的模糊等价矩阵 R, 由该矩阵可得到任一缺陷和其他缺陷的相似度[7,8].

(3) 划分 n 个不同的阈值范围刻画缺陷数据之间的相似程度, 依据模糊等价矩阵中缺陷数据之间的相似度, 依次把缺陷数据分为不同的类, 例如阈值范围为 0.6~0.7, 则相似度在这个范围的分为一类, 其余的单独为一类. 然后分别去除缺陷数据单个属性, 得到不同的模糊等价矩阵, 同样按照阈值范围进行分类.

(4) 通过粗糙集中重要性理论计算属性权重.

1) 设论域 U 是所有缺陷数据的非空有限集合, 粗糙集理论利用近似集的概念来描述缺陷数据各分类簇的粗糙程度. 对 U 中的分类簇 X 及 U 上的一个等价关系 R, 则称集合$R_{-}(X)=\{x\in U:[X]_R\subseteq X\}$为 X 的 R 下近似集, 其本质是删除一个属性后的分类可以准确划分到没有删除任何一个属性的分类中去的缺陷集合, 且必须是原分类集合中的子集. 依据粗糙集理论可把缺陷数据分类结果 D 依赖缺陷所有特征属性 C 的依赖度定义为[9]:

$$\gamma(C,D)=|POS(C,D)|/|U| \tag{5}$$

式中, $POS(C,D)=R_{-}(X)$, $|POS(C,D)|$为正域 POS(C,D)的缺陷个数, 即为能准确划分到原分类中缺陷集合的缺陷个数, $|U|$为缺陷数据的非空有限集合中缺陷个数, C 为缺陷数据的所有特征属性, D 为在所有特征属性下缺陷数据的分类情况. 所以, 在缺陷所有的特征属性之下进行分类, $\gamma(C,D)$=1.

2) 属性的重要性定义为:

$$SGF(f,C,D)=\gamma(C,D)-\gamma(C-\{f\},D) \tag{6}$$

式中, $\gamma(C-\{f\},D)$表示在 C 中缺少属性 f 后, 特征属性对原分类结果的依赖程度.

3) 属性的综合重要性定义为:

$$\delta(C_f)=\left|\sum_{i=1}^{n}\alpha_k\times SGF_{CD\alpha_k}(C_f)\right|/n \tag{7}$$

式中, α_k 为阈值范围的低端取值, n 为设定阈值范围的数量.

对于本文来说全部缺陷均为一类或各个缺陷单独为一类时, 正域的处理方法并不能带来任何信息, 因此本文确定正域集合时将这两种情况排出. 依据各个属性的综合重要性, 计算各个属性的权重:

$$\omega_f=\delta(C_f)/\sum_{f=1}^{n}\delta(C_f) \tag{8}$$

2.2 缺陷数据实例分析

将表 2 中缺陷数据特征属性进行抽象化编码，然后依据 2.1 节(2)中方法建立模糊等价矩阵 R_{ij} 如下.

表 2 软件缺陷数据特征属性

编号	缺陷名称	缺陷类型	测试类型	缺陷状态	来源阶段	严重程度	测试优先级
1	界面信息错误	用户界面	人机交互界面测试	打开	用户	有待改进	一般
2	功能缺失	功能	功能测试	关闭	测试	重要	中级
3	缺少运算符	计算	文档审查	提交	编码	重要	高级
4	操作平台不兼容	接口	互操作测试	关闭	测试	次要	低级
5	计算顺序错误	逻辑	功能测试	验证	设计	重要	高级
6	界面信息错误	用户界面	人机交互界面测试	拒绝	用户	次要	一般
7	编译环境错误	环境	互操作测试	重新打开	编码	次要	次要
8	功能超越	功能	功能测试	打开	设计	次要	低级
9	数据初始化错误	数据	边界测试	不能重现	编码	中等	中级
10	软件执行时间过长	性能	性能测试	不能重现	集成	重要	中级
11	缺少运算符	计算	文档审查	拒绝	编码	严重	高级
12	界面信息不可用	用户界面	人机交互界面测试	修复	集成	次要	低级
13	界面功能布局不符合常规	用户界面	人机交互界面测试	提交	用户	有待改进	一般
14	包含错误的变量检查	逻辑	文档审查	打开	设计	严重	高级
15	计算中等式错误	计算	文档审查	重新打开	编码	重要	中级

$$R_{ij}=\begin{bmatrix} 1 & 0.5461 & 0.4481 & 0.5899 & \cdots & 0.8385 & 0.8684 & 0.5064 & 0.5561 \\ 0.5461 & 1 & 0.4481 & 0.5461 & \cdots & 0.5461 & 0.5461 & 0.5064 & 0.5461 \\ 0.4481 & 0.4481 & 1 & 0.4481 & \cdots & 0.4481 & 0.4481 & 0.4481 & 0.4481 \\ 0.5899 & 0.5461 & 0.4481 & 1 & \cdots & 0.5899 & 0.5899 & 0.5064 & 0.5561 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0.8385 & 0.5461 & 0.4481 & 0.5899 & \cdots & 1 & 0.8385 & 0.5064 & 0.5561 \\ 0.8684 & 0.5461 & 0.4481 & 0.5899 & \cdots & 0.8385 & 1 & 0.5064 & 0.5561 \\ 0.5064 & 0.5064 & 0.4481 & 0.5064 & \cdots & 0.5064 & 0.5064 & 1 & 0.5064 \\ 0.5561 & 0.5461 & 0.4481 & 0.5561 & \cdots & 0.5561 & 0.5561 & 0.5064 & 1 \end{bmatrix}$$

根据模糊等价矩阵可知:

① $0.5<\alpha\le 0.6$, 全部样本分为 1 类: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15};

② $0.6<\alpha\le 0.7$, 全部样本分为 11 类: {1}, {2}, {3, 11}, {4, 7}, {5}, {6}, {8}, {9, 10, 15}, {12}, {13}, {14};

③ $0.7<\alpha\le 0.8$, 全部样本分为 14 类: {1}, {2}, {3}, {4}, {5}, {6}, {7}, {8}, {9, 10}, {11}, {12}, {13}, {14}, {15};

④ $0.8<\alpha\le 0.9$, 全部样本分为 12 类: {1, 6, 12, 13}, {2}, {3}, {4}, {5}, {7}, {8}, {9}, {10}, {11}, {14}, {15};

⑤ $0.9<\alpha\le 1$, 全部样本分为 15 类: {1}, {2}, {3}, {4}, {5}, {6}, {7}, {8}, {9}, {10}, {11}, {12}, {13}, {14}, {15};

删除属性 1(缺陷名称)时, 按照相同方法处理:

① $0.5<\alpha\le 0.6$, 全部样本分为 1 类: {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15};

② $0.6<\alpha\le 0.7$, 全部样本分为 4 类: {1, 4, 6, 7, 8, 12, 13}, {2, 9, 10, 15}, {3, 11, 14}, {5};

③ $0.7<\alpha\le 0.8$, 全部样本分为 13 类: {1}, {2}, {3}, {4, 7}, {5}, {6}, {8}, {9, 10}, {11}, {12}, {13}, {14}, {15};

④ $0.8<\alpha\le 0.9$, 全部样本分为 12 类: {1, 6, 12, 13}, {2}, {3}, {4}, {5}, {7}, {8}, {9}, {10}, {11}, {14}, {15};

⑤ $0.9<\alpha\le 1$, 全部样本分为 15 类: {1}, {2}, {3}, {4}, {5}, {6}, {7}, {8}, {9}, {10}, {11}, {12}, {13}, {14}, {15}.

根据上述属性重要性计算方法, 计算出属性 1 的综合重要性为 0.13; 同样的方法计算出属性 2 的综合重要性为 0.44; 属性 3 的综合重要性为 0.15; 属性 4 的综合重要性为 0.19; 属性 5 的综合重要性为 0.04; 属性 6 的综合重要性为 0.08, 属性 7 的综合重要性为 0.04. 归一化之后得到缺陷数据各个属性的权重值: ω_1=0.12, ω_2=0.41, ω_3=0.14, ω_4=0.18, ω_5=0.04, ω_6=0.07, ω_7=0.04.

3 混合类型属性的相似性计算

对于缺陷数据, 其特征属性包含标称属性和序数属性等, 则需要计算混合类型的属性相似性.

假设数据集包含 p 个混合类型的属性, 对象 i 和 j 之间的相似性 $sim(i,j)$ 定义为:

$$sim(i,j)=1-\sum_{f=1}^{p}\omega_f d_{ij}^{(f)} \tag{9}$$

其中, ω_f 是属性 f 的权重; $d_{ij}^{(f)}$ 是缺陷 i,j 对于属性 f 的相异度. 属性 f 对 i 和 j 之间的相似度按照相应属性类型计算:

(1) 当属性 f 是序数属性时, $d_{ij}^{(f)}=d(i,j)$.

(2) 当属性 f 是标称属性时, 如果 $x_{if}=x_{jf}$ 时, $d_{ij}^{(f)}=0$, 否则 $d_{ij}^{(f)}=1$.

上述方法获得的相似矩阵 $sim(i,j)$, 其中元素分别代表 i 和 j 两个缺陷之间的相似性, 能够直接反映缺陷数据之间的相似特征.

分析上述对 15 个缺陷数据进行分析的权重结果, 假设缺陷数据仅有表 1 中三个属性表征时的权重分别为 0.66, 0.66, 0.22, 0.12. 利用混合属性的相似性计算方法, 则缺陷 1 和缺陷 4 的相似度为:

$$Sim(1,4)=1-\sum_{f=1}^{p}\omega_f d_{ij}^{(f)}=$$
$$1-[0.66\times(0)+0.22\times(1)+0.12\times(0.33)]=0.74.$$

由此我们可以得到相似度矩阵:

$$Sim(i,j)=\begin{bmatrix} 1 & & & \\ 0.04 & 1 & & \\ 0 & 0.08 & 1 & \\ 0.74 & 0.08 & 0.04 & 1 \end{bmatrix}$$

4 实验结果及分析

为了体现本方法的通用性, 同时便于与已有的研究成果进行比较, 本文选用 XXX 单位通用装备保障软件测评中心和软件工程实验室近 3 年来经过软件测试获得的 400 个软件缺陷数据进行分析, 该数据主要包括缺陷编号, 状态, 产品名称, 版本号, 优先级, 描述, 评论等信息. 该数据均为信息管理类软件缺陷, 只是其中各个对象有一定的不同, 所以测试数据之间有一定的相似性. 该数据来源于实验室中心实际测评, 数据类型比较丰富, 包含内容比较全面, 具有一定的真实性, 应用于本文方法进行分析比较适合且其结果具有一定的说服性.

不同方法的相似性度量量化方法普遍采用查全率(Recallrate)、查准率(PercisionRate)作为衡量指标, 其定义如下:

$$RecallRate=\frac{N_{detected}}{N_{total}} \tag{10}$$

$$PercisionRate=\frac{N_{detected}}{N_{detectedall}} \tag{11}$$

其中, $N_{detected}$ 是指实验检测到的相似缺陷数据中正确的个数, N_{total} 是指测试缺陷数据实际包含的相似缺陷数据的个数, $N_{detectedall}$ 是指实验检测到的所有相似缺陷数据的总数. VSM 向量空间模型的相似性度量方法主要是通过缺陷报告中的概要描述和详细描述, 提取特征项, 进行文本的相似度计算, 将本文方法和 VSM 方法进行比较, 结果分析如表 3 和表 4 所示.

表 3 两种方法查准率对比实验结果

衡量指标	阈值		
	0.4	0.5	0.6
本文查准率	68	87	94
VSM查准率	64	81	91

表 4 两种方法查全率对比实验结果

衡量指标	阈值		
	0.4	0.5	0.6
本文查全率	85	76	61
VSM查全率	78	61	43

从表 3, 表 4 可以看出, 本文基于模糊聚类的相似性度量方法在查准率和查全率上比 VSM 向量空间模型的相似性度量方法效果有一定程度的提升, VSM 方法通过软件缺陷报告和缺陷的详细描述等进行数据样本的特征项提取, 然后进行文本相似度计算. VSM 方法计算抽象化特征项的相似距离时, 取值比较小的特征值被取值较大的特征值淹没, 导致计算结果不准确.

5 结论

本文采用的相似性度量方法一方面引用属性权重的重要性, 基于已有的缺陷数据分类情况计算提取的缺陷特征属性权重, 另一方面将缺陷属性进行相似性分析, 并解决了抽象化数据时所定义的属性值大小对相似度的影响, 避免了较小数值被淹没的现象. 因此, 本文方法相对于基于 VSM 的相似度分析方法具有更高的稳定性和计算效果. 本文方法提出缺陷属性权重的计算方法, 并且基于数据挖掘算法提出相似度的计算公式. 在下一步工作中, 需要引入更简便的权重计算方法, 或者在不区分属性类别的情况下引入相应系数调整相似度结果, 以此提高本文方法的效率, 降低计算复杂度.

参考文献

1 王骏, 王士同, 邓赵红. 聚类分析研究中的若干问题. 控制与决策, 2012, 27(3): 321–328.
2 王斌, 吴太文, 胡培培. 软件缺陷分类和分析研究. 计算机科学, 2013, 40(9): 16–20, 24.
3 Han JW, Kamber M, Pei J. 数据挖掘概念与技术. 范明, 孟小峰, 译. 3 版. 北京: 机械工业出版社, 2012.
4 黄定轩, 武振业, 宗蕴璋. 基于属性重要性的多属性客观权重分配方法. 系统工程理论方法应用, 2004, 13(3): 203–207.
5 刘文军. 连续值域决策表的一种属性权重确定方法. 模糊系统与数学, 2008, 22(3): 160–166.
6 曹秀英, 梁静国. 基于粗集理论的属性权重确定方法. 中国管理科学, 2002, 10(5): 98–100.
7 柳炳祥, 李海林. 基于模糊粗糙集的因素权重分配方法. 控制与决策, 2007, 22(12): 1437–1440.
8 杨淑莹, 张桦. 模式识别与智能计算—MATLAB 技术实现. 3 版. 北京: 电子工业出版社, 2015.
9 李秀格. 基于模糊等价矩阵的模糊聚类相关理论研究[硕士学位论文]. 沈阳: 辽宁大学, 2015.
10 李楠, 王晓博, 刘超. 自动分析软件缺陷报告间相关性的方法研究. 计算机应用研究, 2010, 27(6): 2134–2139.