

基于加权 RFE-Bayes 方法的软件缺陷预测模型

梁天超^{1,2}, 荆晓远², 姚永芳², 董西伟²

(1. 南京邮电大学 计算机学院, 江苏 南京 210003;

2. 南京邮电大学 自动化学院, 江苏 南京 210003)

摘要:近年来,软件缺陷预测逐渐成为软件工程领域的重要内容。很多典型的机器学习方法已经被应用到软件缺陷预测中,包括 SVM、随机森林、决策树和朴素贝叶斯等。早期的研究工作对所有软件产品采取相同的特征提取方式,分类效果并不理想。后来一些特征选择方法被提出,比如基于启发式的回归特征消除方法已经成功与 SVM 方法结合起来,取得了较好的效果。文中在现有工作基础上借鉴了 RFE(回归特征消除)的思想,考虑到朴素贝叶斯方法在处理小样本分类问题时的优越性,首次将 RFE 与朴素贝叶斯方法结合起来,利用贝叶斯模型的特性在特征选择后将特征权值应用到对分类决策的改进中,进一步提高了分类器性能。实验采用 NASA 的软件缺陷数据集,并对比其他效果较好的分类算法,体现了该算法的优越性和有效性。

关键词:软件缺陷;特征选择;朴素贝叶斯;缺陷预测

中图分类号:TP31

文献标识码:A

文章编号:1673-629X(2015)10-0131-04

doi:10.3969/j.issn.1673-629X.2015.10.029

A Prediction Model for Software Defect Based on Weighted RFE-Bayes

LIANG Tian-chao^{1,2}, JING Xiao-yuan², YAO Yong-fang², DONG Xi-wei²

(1. College of Computer, Nanjing University of Posts and Telecommunications,
Nanjing 210003, China;

2. College of Automation, Nanjing University of Posts and Telecommunications,
Nanjing 210003, China)

Abstract: In recent years, software defect prediction is becoming an important part of the software engineering field. Many typical methods like SVM, random forest, decision trees and Bayes have been applied to software defect prediction. However, earlier research almost takes the same feature set to train all kinds of software products and does not achieve a desired effect. Years later, some feature selection method are proposed. For example, the method combined recursive feature elimination and SVM, has got a good effect. In this paper, based on existing work, propose an algorithm which combines recursive feature elimination and Native Bayes. This algorithm will do a selection of feature set before training the model according to the contribution of each feature to get the optimal feature subset to be the input to train the model. The experiment adopts the software defect data set of NASA. Make a comparison with other machine learning methods, the experimental results demonstrate the superiority and effectiveness of this method.

Key words: software defect; feature selection; Naive Bayes; defect prediction

1 概述

软件缺陷预测是一种在软件开发过程中对软件进行度量、统计、分析以对潜在的软件缺陷进行预测的技术。有效的缺陷预测有助于更合理地安排测试资源,评估软件质量^[1-3]。

软件缺陷预测实际是一个二分类问题:将软件模

块判定为有缺陷或没有缺陷。在软件工程中,通过对历史版本中有缺陷标注的软件缺陷模块进行代码属性度量,采用统计和机器学习的方法对软件模块进行分析是当前软件缺陷预测的主流做法。目前最为广泛的属性度量方法是静态代码度量,其指标为:Halstead^[4]和 McCabe^[5]。常用的机器学习分类算法如支持向量

收稿日期:2015-01-29

修回日期:2015-05-05

网络出版时间:2015-09-23

基金项目:国家自然科学基金资助项目(61272273);江苏省333工程项目(BRA2011175);南京邮电大学科研项目(XJKY14016)

作者简介:梁天超(1989-),男,硕士,研究方向为计算机取证技术;荆晓远,教授,研究方向为模式识别、图像处理、智能信息处理、机器学习等。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20150923.1505.036.html>

机^[6]、决策树^[7]和朴素贝叶斯^[8]等都已经用来构建软件缺陷预测模型,而且取得了一定的分类效果。然而,早期的软件缺陷检测方法对所有软件产品采用相同的特征提取方式,训练出的分类模型泛化能力较低。近年来,多种特征选择方法已经被提出,并与主流的机器学习方法结合起来。

特征选择是为了在训练分类器前减少特征维度,以达到降低运算复杂度和去除冗余特征的目的。当前特征选择的搜索策略和评价准则多种多样,常用的特征选择搜索策略有:随机搜索策略、启发式搜索策略(如序列前向选择策略、序列后向选择策略)等。目前基于启发式的搜索策略越来越受到人们的关注,其避免了随机搜索的高耗时情况,在较少时间内选出最优特征集合。借鉴特征后向选择思路,Wang 等提出了一种基于支持向量机最大间隔原理的 RFE-SVM 方法,该方法通过学习样本特征的权重对 SVM 分类器进行降维,是目前最有效的特征选择分类方法之一^[9]。但是,该方法并没有将学习到的特征权重应用到分类决策中,考虑到在最优特征子集中各个特征对分类结果的影响应该是有差异的,该算法仍有较大的改进空间。

文中的主要贡献有:

(1)第一次将有着坚实的数学基础和稳定的分类效率的朴素贝叶斯方法与 RFE(循环特征消除)方法结合起来,缩短了预测模型的复杂度,并加强了泛化能力;

(2)提出了一种在特征选择的基础上对最优特征集进行加权的加权 RFE-Bayes 方法,进一步提高了模型的预测能力。实验分析中对比了此方法与其他传统机器学习方法如 RFE-SVM 方法的分类性能,表现出了一定的优势。

2 相关方法介绍

2.1 朴素贝叶斯模型

很多统计学和机器学习中的方法已经被应用到软件缺陷预测的工作中^[10-11]。机器学习方法可以高效地利用少量信息进行分类。其可以通过学习手段获得一个结构化的预测模型,根据软件模块的相关特征来对其是否具有缺陷进行分类。很多数据挖掘手段如决策树、随机森林、SVM 和贝叶斯以及一些其他的技术手段已经在软件缺陷检测中被广泛应用。软件缺陷领域存在小样本问题,即软件模块的标记(有缺陷或无缺陷)需要人为统计,在有限的时间和资源条件下难以获得大量样本。在此条件下,朴素贝叶斯算法表现出了优于其他机器学习方法的性能。

朴素贝叶斯是一种高效的统计学分类算法,并且

已经被成功地应用到生物信息学领域^[12]。朴素贝叶斯的基本理论类似于协方差行列式,但是对朴素贝叶斯来讲,其要立足于一个基本假设:即每个属性变量对结果的影响都是独立的,这一假设大大简化了条件概率的计算。在朴素贝叶斯算法框架中,一个分类问题可以被看作在已知一组观察变量的基础上找到最大概率出现的结果的问题。以软件缺陷检测为例:样本具有的属性为 $F = \{f_1, f_2, \dots, f_n\}$, 而试图找到一个类别 C , 使 $P(F | C) = P(f_1, f_2, \dots, f_n | C)$ 尽可能取到最大值。由于当前的目标是试图区分软件缺陷模块和无缺陷模块,因此可以推出二值类 $C \in \{0, 1\}$ 。其中 1 代表细胞样本被预测为有缺陷模块,0 代表样本被预测为无缺陷模块。

对于这样一个二分类问题来说,软件模块的分类结果可以由比较两个概率值来确定:

$$\frac{P(C = 1 | F = f_1, f_2, \dots, f_n)}{P(C = 0 | F = f_1, f_2, \dots, f_n)} = \frac{P(C = 1) \prod_{i=1}^n P_i(f_i | C = 1)}{P(C = 0) \prod_{i=1}^n P_i(f_i | C = 0)} \quad (1)$$

对式(1)取对数,可以得到:

$$\log \frac{P(C = 1 | F = f_1, f_2, \dots, f_n)}{P(C = 0 | F = f_1, f_2, \dots, f_n)} = \log \frac{P(C = 1)}{P(C = 0)} + \sum_{i=1}^n \log \frac{P_i(f_i | C = 1)}{P_i(f_i | C = 0)} \quad (2)$$

因此在以下条件时样本将被预测为 1(有缺陷模块):

$$\log \frac{P(C = 1 | F = f_1, f_2, \dots, f_n)}{P(C = 0 | F = f_1, f_2, \dots, f_n)} \geq \theta \quad (3)$$

反之则预测为 0(无缺陷模块)。其中, θ 是用来折衷灵敏度和特异性的阈值。

2.2 特征选择

通过机器学习对数据进行分类一般包括两个过程:特征提取和分类器学习。为了针对不同类型的软件产品得到高性能的分类器,选取合适的特征子集非常重要^[13]。在之前的软件缺陷检测研究中,通常把研究重点放在分类器的学习上,而对所有软件产品都采取统一的特征提取方式,不在此基础上对提取的特征进行进一步的选择。近年来,一些特征选择方法相继被提出,一般的特征选择过程为:先从初始的特征集中产生一个特征子集,然后使用一种合理的评价函数对这个特征子集进行评价,评价结果将被与停止准则相比较,如果达到停止标准则退出,否则进行下一轮特征选择,重复这个过程直到选出合适的特征子集为止。也就是说,特征选择的过程主要包括以下四个部分:子集的产生过程,评价过程,停止准则和验证过程。RFE 为基于序列后向选择的启发式搜索算法,基本流程如图 1 所示。

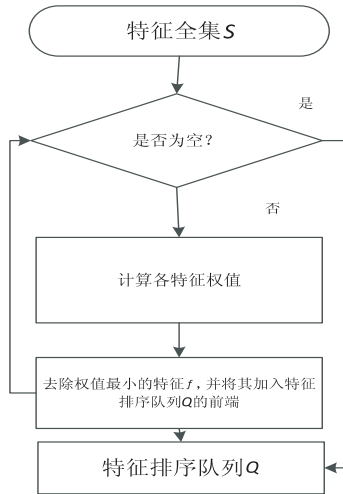


图 1 RFE 特征选择过程

3 特征加权 RFE-Bayes 方法

3.1 Naive Bayes 属性加权

文中对特征的筛选使用的是 RFE (Recursive Feature Elimination, 回归特征消除) 思想, 寻找最优子集。基本思想是基于序列后向选择算法, 从特征全集 O 开始, 每次从特征全集 O 剔除一个权值最小的特征 x , 使得提出特征 x 后评价函数值达到最优。因此, 不重要或者与分类相关性小的特征会被排在特征排序列表的最后位置, 并在特征筛选的时候优先被排除, 最重要的特征则会被留到最后。

在特征排序列表中, 特征被选择的数目会受到评估标准的影响。解决 Bayes 分类问题时, 特征权值的大小可以反映出其对分类决策的贡献程度, 因此只要得到特征相应的权值, 既可作为特征排序的依据。每次删除最不重要的特征后对当前特征子集进行预测精度验证, 根据分类器的性能表现来评估该子集的好坏。将上述所有特征子集全部评估完毕后可以得到最优的特征子集。权值可以由每种特征的先验概率正确性来得到。由式(1)可知, 对一个样本 x 的分类结果实际上由 x 在不同类别条件下每个特征 f_i 出现的概率相积决定的。对于每个特征都会对 x 的类别进行一次独立的判断。那么在预测过程中通过每种特征判断的准确度即可得到该特征对所有样本分类做出贡献的总体情况。因此, 文中设置权值为:

$$\omega_j = \sum_i \varphi(f_{ij}, P, y_i) \quad (4)$$

其中, ω 是特征的权值向量; $\varphi(x_i, P, y_i)$ 为第 i 个样本对特征权值的贡献度, 具体的:

$$\varphi(x_i, P, y_i) = \begin{cases} 1 & \text{如果该属性对 } x_i \text{ 的判别与 } x_i \text{ 实际类别相同} \\ 0 & \text{其他} \end{cases}$$

由此得到的每个特征的权值 ω 即可表征其预测能

力的准确性。

3.2 加权 RFE-Bayes 算法描述

文中提出的 RFE-Bayes (Recursive Feature Elimination based Naive Bayes, 基于贝叶斯的回归特征消除) 是利用 RFE 进行特征提取, 同时借助 Bayes 进行分类的过程。RFE-Bayes 借鉴后向消除思想^[14], 在达到停止准则前不断进行迭代, 每次迭代包括以下三个步骤:

(1) 使用初始训练样本集训练 Bayes 分类器, 得到每个属性的先验概率矩阵 P ;

(2) 通过先验概率矩阵和训练样本来得出每个属性的权值 ω ;

(3) 从当前的特征集中移除权值最小的特征。

循环上述过程直到数据集达到所定义的停止准则时结束, 执行的结果是根据所定义的算法得到该子集中特征的重要性排序。文中把剔除后使分类器性能下降的特征定义为“正特征”, 剔除后使分类器性能上升的特征为“负特征”。根据特征排序表的结果, 一次将排序靠前的“正特征”加入到空集 \emptyset 中, 获得子集 $F_1 \subset F_2 \subset \dots \subset F_k$, 并以 Bayes 分类器的预测结果来评估这些子集的性能, 从而获得最优子集。设训练样本为 $X_n = [x_1, x_2, \dots, x_n]^T$, 其对应的标记向量为 $Y_n = [y_1, y_2, \dots, y_n]^T$; 初始向量集为 S , 特征排序向量 r 为空集 \emptyset , 加权 RFE-Bayes 的训练过程如下:

Step1: Repeat until $S = \emptyset$:

(1) sample matrix $X = X_0(:, S)$

(2) Train model $\alpha = \text{Bayes} - \text{train}(X, Y)$

(3) Compute weight value $\omega_j = \sum_i \varphi(f_{ij}, P, y_i)$

(4) Find the feature with min weight value $f = \text{argmin}(\omega)$

(5) Update feature order vector $r = [s(f), r]$

(6) Delete f ; $S = S(1:f-1, f+1:\text{length}(s))$

Step2: For every +feature f in r :

$S_j = (S, f)$

$S' = \text{optimal}(S_j)$

Step3: train model M with S'

Output: M

每得到一个新的特征子集, 就要通过训练得出相应的权值向量 W , 并对其进行归一化处理。在分类时, 考虑到不同属性的权重影响, 式(1)改写为:

$$\log \frac{\prod_{i=1}^n P(C=1 | F=f_i)^{\omega_i}}{\prod_{i=1}^n P(C=0 | F=f_i)^{\omega_i}} \geq \theta \quad (5)$$

4 实验

4.1 数据库介绍

为比较模型性能, 从 NASA 数据库中挑选 5 个软

件缺陷数据集进行试验。数据集特征如表 1 所示。对于每种数据集选取了现行基于机器学习的软件缺陷预测中预测性能比较好的几种算法,包括随机森林(Random Forest, RF)、决策树(Decision Tree, DT)、C4.5、SVM 和未进行特征选择的朴素贝叶斯(Naive Bayes, NB)。

表 1 数据集信息

| 数据集 | 语言 | 属性数 | 样本数 | 缺陷数 |
|-----|------|-----|-------|-----|
| MC2 | C++ | 40 | 155 | 51 |
| PC1 | C | 22 | 919 | 60 |
| PC3 | Java | 38 | 1 409 | 148 |
| PC4 | C++ | 38 | 1 270 | 176 |
| PC5 | C | 39 | 1 694 | 457 |

4.2 评价指标

对分类模型的评价指标包含预测率(P_d)、误报率(P_f)、准确率(P)和召回率 R 。表达式如下:

$$P_d = \frac{TP}{TP + FN}$$

$$P_f = \frac{FP}{TN + FP}$$

$$P = \frac{TP}{TP + FP}$$

其中,TP 代表被正确识别为缺陷模块的软件模块数;TN 代表被正确识别为无缺陷的软件模块数;FP 代表被认定为有缺陷的实际上无缺陷的模块;FN 代表被识别为无缺陷的而实际上存在缺陷的模块。在此基础上定义 F 值^[15-16](F -measure):

$$F = \frac{2 * P * R}{P + R} \quad (6)$$

其中, F 值表征着准确率和召回率的加权调和平均,反映了准确率和召回率的综合情况。

在软件缺陷预测领域, F 值是判断预测模型性能的重要观察指标。

4.3 实验结果与分析

文中在 NASA 的 5 个数据库上进行了实验,实验结果表明各个方法在不同数据集上的性能比较稳定,而在相同数据集上的不同算法模型表现出较大的性能差异,对预测率和误报率实验结果如表 2~4 所示。

表 2 各方法的 P_d 比较

| 评价指标 | 数据集 | C4.5 | RFE-SVM | Bayes | RFE-Bayes |
|-------|-----|--------|---------|--------|-----------|
| P_d | MC2 | 44.793 | 45.81 | 71.379 | 78.516 |
| | PC1 | 34.407 | 59.76 | 48.627 | 53.489 |
| | PC3 | 30.195 | 36.954 | 58.563 | 64.419 |
| | PC4 | 44.748 | 57.402 | 59.508 | 65.458 |
| | PC5 | 45.288 | 63.918 | 70.929 | 78.021 |

表 3 各方法的 P_f 比较

| 评价指标 | 数据集 | C4.5 | RFE-SVM | Bayes | RFE-Bayes |
|-------|-----|--------|---------|--------|-----------|
| P_f | MC2 | 25.281 | 21.816 | 48.96 | 46.512 |
| | PC1 | 7.812 | 17.037 | 15.138 | 14.381 |
| | PC3 | 7.236 | 37.161 | 22.338 | 21.221 |
| | PC4 | 6.876 | 14.823 | 16.299 | 15.484 |
| | PC5 | 1.584 | 20.025 | 7.047 | 6.694 |

表 4 各方法的 F -measure 比较

| 评价指标 | 数据集 | C4.5 | RFE-SVM | Bayes | RFE-Bayes |
|--------------|-----|--------|---------|--------|-----------|
| F -measure | MC2 | 45.837 | 50.328 | 43.353 | 54.877 |
| | PC1 | 31.446 | 28.557 | 28.872 | 32.84 |
| | PC3 | 25.335 | 34.398 | 26.748 | 39.557 |
| | PC4 | 41.931 | 41.022 | 44.172 | 44.175 |
| | PC5 | 14.643 | 32.832 | 43.488 | 37.756 |

通过分析上述结果,可以得出如下结论:

(1)C4.5 决策树的预测率较低,并且对于类不平衡越严重的数据集,C4.5 决策树与其他方法的差距越来越明显。不过这种算法的误报率也比较低,这也和预测率低有关。

(2)RFE-SVM 算法和朴素贝叶斯算法表现出了较为优异的性能,并且在样本数量明显较少的 MC2 数据集上也表现出了较强的分类能力。

(3)加权 RFE-Bayes 算法较传统的朴素贝叶斯算法来说误报率略有提高,而换来的是预测率的明显提高。说明了文中提出的特征选择方法的有效性。

(4) F -measure 代表着预测率和召回率的综合情况。通过比较可知,文中提出的加权 RFE-Bayes 算法在 F -measure 上要优于 RFE-SVM 方法,体现了对特征加权的有效性。

5 结束语

文中考虑到不同软件特征的差异性,提出一种基于特征选择的贝叶斯分类模型,其对不同的软件样本提取不同的特征进行训练和测试,并根据特征权值调整特征集中不同特征在分类时的作用,提高了贝叶斯模型的泛化能力。实验结果表明,文中方法显著提高了传统贝叶斯方法的分类性能,与现有的其他基于特征选择的软件缺陷预测方法相比也具有一定的优势。

参考文献:

- [1] 宋颖. 面向嵌入式 C 的软件缺陷检测技术的研究与实现[D]. 北京:北京邮电大学,2011.
- [2] 朱雪梅. 基于动态方法的嵌入式软件缺陷检测技术研究

时用户的易用性也受到影响,如何在安全性和易用性上实现统一将是一个未来值得研究的方向。


参考文献:

- [1] Top 8 operating systems from Nov 2013 to Nov 2014 [EB/OL]. [2013]. <http://gs.statcounter.com/#all-os-ww-monthly-201311-201411>.
 - [2] Clickjacking [EB/OL]. [2014]. <https://www.owasp.org/index.php/Clickjacking>.
 - [3] Iframe content background defaults to transparent [EB/OL]. [2013]. https://bugzilla.mozilla.org/show_bug.cgi?id=154957.
 - [4] SecTheory. Clickjacking [EB/OL]. [2013]. <http://www.sectheory.com/clickjacking.htm>.
 - [5] Stone P. Next generation clickjacking [R/OL]. 2010. http://www.contextis.com/documents/5/Context-Clickjacking-white_paper.pdf.
 - [6] Niemietz M. Ui redressing: attacks and countermeasures revisited [R/OL]. 2011. <http://ui-redressing.mniemietz.de/uiRedressing.pdf>.
 - [7] Huang L S, Moshchuk A, Wang H J, et al. Clickjacking: attacks and defenses [C]//Proc of USENIX security symposium. [s. l.]: USENIX Association, 2012: 413-428.
 - [8] Rydstedt G, Gourdin B, Bursztein E, et al. Framing attacks on smart phones and dumb routers: tap-jacking and geo-localization attacks [C]//Proceedings of the 4th USENIX conference on offensive technologies. [s. l.]: USENIX Association, 2010: 1-8.
 - [9] Luo T, Jin X, Ananthanarayanan A, et al. Touchjacking attacks on web in Android, IOS, and windows phone [M]//Foundations and practice of security. Berlin: Springer, 2013: 227-243.
 - [10] Android. webkit | Android developers [EB/OL]. [2014]. <http://developer.android.com/reference/android/webkit/package-summary.html>.
 - [11] IE8 security part VII: ClickJacking defenses [EB/OL]. [2010]. <http://blogs.msdn.com/b/ie/archive/2009/01/27/ie8-security-part-vii-clickjacking-defenses.aspx>.
 - [12] Rydstedt G, Bursztein E, Boneh D, et al. Busting frame busting: a study of clickjacking vulnerabilities at popular sites [J]. IEEE Oakland Web, 2010, 2: 6-15.
 - [13] Luo T, Jin X, Du W. Mediums: visual integrity preserving framework [C]//Proceedings of the third ACM conference on data and application security and privacy. [s. l.]: ACM, 2013: 309-316.
 - [14] Enck W, Ongtang M, McDaniel P. Understanding android security [J]. IEEE Security & Privacy, 2009, 7(1): 50-57.
-
- (上接第 134 页)
- 实现 [D]. 杭州: 杭州电子科技大学, 2014.
 - [3] 尹相乐, 马力, 关昕. 软件缺陷分类的研究 [J]. 计算机工程与设计, 2008, 29(19): 4910-4913.
 - [4] Li M, Kang H, Zhou P, et al. Hybrid optimization algorithm based on chaos, cloud and particle swarm optimization algorithm [J]. Journal of Systems Engineering and Electronics, 2013, 24(2): 324-334.
 - [5] Zeng G P, Fan H L. Two-subpopulation particle swarm optimization based on pheromone diffusion [J]. Applied Mechanics and Materials, 2014, 667: 300-308.
 - [6] Vapnik V N. The nature of statistical learning theory [M]. New York: Springer-Verlag, 1995.
 - [7] Basili V, Green S, Laitenberger O, et al. The empirical investigation of perspective-based reading [J]. Empirical Software Engineering, 1996, 1: 133-164.
 - [8] Možina M, Žabkar J, Bratko I. Argument based machine learning [J]. Artificial Intelligence, 2007, 171: 922-937.
 - [9] Mundra P A, Rajapakse J C. SVM-RFE with MRMR filter for gene selection [J]. IEEE Trans on NanoBioscience, 2010, 9(1): 31-37.
 - [10] 王青, 伍书剑, 李明树. 软件缺陷预测技术 [J]. 软件学报, 2008, 19(7): 1565-1580.
 - [11] Kazman R, Bass L, Abowd G, et al. SAAM: a method for analyzing the properties of software architectures [C]//Proceedings of the 16th international conference on software engineering. Sorrento, Italy: IEEE, 1994: 81-90.
 - [12] 葛贺贺, 金聪, 叶俊民. 基于 PSO 和朴素贝叶斯的软件缺陷预测模型 [J]. 计算机工程, 2011, 37(12): 36-37.
 - [13] Whatling C, McPheat W, Hersloef M. The potential link between atherosclerosis and the 5-lipoxygenase pathway: investigational agents with new implications for the cardiovascular field [J]. Expert Opinion on Investigational Drugs, 2007, 16(12): 1879-1893.
 - [14] Johannes M, Brase J C, Fröhlich H, et al. Integration of pathway knowledge into a reweighted recursive feature elimination approach for risk stratification of cancer patients [J]. Bioinformatics, 2010, 26(17): 2136-2144.
 - [15] Goulão M, Fonte N, Wermelinger M, et al. Software evolution prediction using seasonal time analysis: a comparative study [C]//Proc of 16th European conference on software maintenance and reengineering. [s. l.]: IEEE, 2012: 213-222.
 - [16] Fawcett T. An introduction to ROC analysis [J]. Pattern Recognition Letters, 2006, 27(8): 861-874.

基于加权RFE-Bayes方法的软件缺陷预测模型

作者: [梁天超](#), [荆晓远](#), [姚永芳](#), [董西伟](#), [LIANG Tian-chao](#), [JING Xiao-yuan](#), [YAO Yong-fang](#), [DONG Xi-wei](#)

作者单位: [梁天超, LIANG Tian-chao\(南京邮电大学 计算机学院, 江苏 南京 210003; 南京邮电大学 自动化学院, 江苏 南京 210003\)](#), [荆晓远, 姚永芳, 董西伟, JING Xiao-yuan, YAO Yong-fang, DONG Xi-wei\(南京邮电大学 自动化学院, 江苏 南京, 210003\)](#)

刊名: [计算机技术与发展](#) 

英文刊名: [Computer Technology and Development](#)

年, 卷(期): 2015(10)

引用本文格式: [梁天超](#). [荆晓远](#). [姚永芳](#). [董西伟](#). [LIANG Tian-chao](#). [JING Xiao-yuan](#). [YAO Yong-fang](#). [DONG Xi-wei](#)

[基于加权RFE-Bayes方法的软件缺陷预测模型](#) [期刊论文] - [计算机技术与发展](#) 2015(10)