|  |
| --- |
| **中国地质大学课程报告** |
| **《**软件缺陷预测研究综述**》**  **学 号： 20191001171**  **姓 名： 滕德淋**  **学 科 专 业： 软件工程**  **指 导 教 师： 刘袁缘**  **培 养 单 位： 地理与信息工程学院** |

|  |
| --- |
| 二○二一年 十一月 |

摘要

软件缺陷预测的目的是在展开软件测试之前，利用软件相关的一些属性建立模型判别具体的软件模块是否含有缺陷或含有缺陷的数目，为软件测试的资源分配提供决策支持。近年来，软件缺陷预测技术的研究吸引了大量学者和软件从业者的关注，同时也产生了大量的研究成果，因此有必要对软件缺陷预测相关的成果进行系统性的梳理。首先简要介绍了软件缺陷预测的概念，接着分析了常用数据集，并对缺陷度量元的研究及建模方法的研究现状进行了阐述，最后总结了全文。

**关键词**：软件缺陷预测；缺陷数据集；软件度量元；机器学习；分类

**Abstract**

The purpose of software defect prediction is to establish a model using known software-related attributes to determine whether a specific software module contains a defect or the number of defects it contains before software testing is started, and to provide decision support for software testing resources allocation. In recent years, many researchers and software practitioners have paid attention to the research on software defect prediction techniques, and has also produced a large number of research results. Therefore, it is necessary to systematically sort out the results related to software defect prediction. This article first briefly introduces the concept of software defect prediction, then analyzes commonly used data sets, describes the research status of defect metric units and the research status of modeling methods, finally concluded.

**Key Words:** Software defects Prediction; Defect dataset; Software metrics; Machine learning; Classification

**目 录**

目录

[第一章 软件缺陷预测 4](#_Toc87724537)

[第二章 缺陷数据集 4](#_Toc87724538)

[第三章 度量元的研究 5](#_Toc87724539)

[第四章 模型的构建 6](#_Toc87724540)

[4.1 基于分类的技术 6](#_Toc87724541)

[4.2 基于回归的技术 6](#_Toc87724542)

[4.3 基于聚类的技术 7](#_Toc87724543)

[第五章 结语 7](#_Toc87724544)

[参考文献 8](#_Toc87724545)

# 软件缺陷预测

软件缺陷是软件中的故障、错误或者失效，它的存在会使软件系统产生意外的结果或者出现行为异常，导致软件产品无法正常运行。如果能够对易发缺陷组件进行准确的预判，将有助于软件测试工作，降低软件开发成本，在测试工作中重点关注易发缺陷组件，从而提高软件产品质量。软件缺陷预测则是一种对软件易发缺陷组件进行预测的技术，在此方向上，学者重点关注缺陷倾向性、缺陷数目及缺陷严重性的研究，主要关注点在缺陷数据集、缺陷模型的构建、缺陷度量元以及模型评价等方面。

# 缺陷数据集

经过几十年的发展积累，软件缺陷预测领域已有不少可用于数据建模的数据集，基于对已有文献的调查，当前使用较多的数据集有如下几种：

1. NASA[NASA数据集：https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/]数据集：此数据集是美国宇航局公开的，公众可以免费使用的公共数据库，此库包含了美国宇航局对其多个软件项目的统计信息，它是现目前研究中使用最广泛的数据集。
2. PROMISE[PROMISE数据集：https://code.google.com/p/promisedata/]数据集：这些数据可在PROMISE库中免费公开获得，其中包括Eclipse集、Jedit软件数据集以及其他众多开源数据集。
3. AEEEM数据集：此数据集由D’Ambros等人整理,共包含5个不同的开源项目。
4. SoftLab数据集：此数据集来源于3个Turkish软件公司。
5. 开源社区数据集：如SourceForge, Github, StackOver-flow等大型开源社区为研究者提供了软件问答、软件版本等信息，提供了大量的特征，这些特征能够用于软件相关属性与软件缺陷之间关系的挖掘，这也为软件缺陷研究的数据来源提供了新的可能。

虽然已有众多的数据集为软件缺陷预测的研究提供支持，但是在使用的过程中也遇到了一些问题。比如来自于私人或公司的数据集在使用的过程中涉及隐私问题而无法对他人基于该数据集取得的成果进行复现，公开数据集缺乏有效标注等。

此外，软件缺陷数据集通常面临以下三类问题：类不平衡、高维度以及噪音问题。数据类不平衡是指在日常的软件中，软件模块缺陷的分布大致满足二八定律：即80%的软件缺陷分布在20%软件模块当中。因此在实际的软件产品中，无缺陷模块远远多于有缺陷模块，影响了分类学习器对少数类（即有缺陷模块）的预测精度。缺陷数据的高维度是指数据集中有大量可用于预测建模的度量元，在建模时选用过多的特征会造成信息的冗余，因为部分特征与特征之间的信息存在重复或者包含关系，从而带来数据的维数灾难，维数灾难不仅加重了建模的时间成本，同时也降低了模型的预测性能。对软件项目进行标注或者度量的工作是人工完成的，因此在此过程中易出现一些错误，此类错误为数据的噪音问题，噪音问题的存在对模型的有效性构成威胁。

# 度量元的研究

为了保证软件质量，开发人员引入了软件度量元用于估计当前正在开发的软件组件的质量，通过预先定义的度量标准对软件组件进行定量分析。在软件工程出现之初设计的度量元主要有规模度量元和复杂度度量元，随着面向对象开发技术的发展，一些面向对象的度量指标被提出，其中最常见的是C&K度量和MOOD度量。在软件的开发周期中，软件组件会不断被更新，在代码变更的过程中可能会引入一些新的缺陷，因此，学者研究设计了一些基于代码变更的度量元。文献[8]采用广泛使用的NR、NML、NDC和NDPV度量构建软件缺陷预测模型。软件开发过程是由人来主导的，所以出现了基于开发者的度量元，基于开发人员的指标主要收集开发人员的经验、代码所有权、编码技能、提交活动等。

# 模型的构建

自缺陷预测广受关注以来，大量缺陷预测模型被构建，但近年来，学者将建模兴趣集中在了机器学习上，机器学习被广泛应用于软件缺陷模型的研究，目前研究中使用的机器学习方法主要有：Logistic回归（Logistic Regression，LR）、决策树（Decision Trees，DT）、随机森林（Random Forest，RF）、朴素贝叶斯（Naive Bayes，NB）、支持向量机（Support Vector Machine， SVM）、神经网络（Neural Networks，NN)）、集成学习（Ensemble Learning，EM）等，接下来，将对这些方法进行总结分析。

## 4.1 基于分类的技术

文献[9]使用决策树和神经网络方法构建模型预测不同严重程度的故障分类，基于公开数据集NASA数据集进行研究得出机器学习方法对所有严重程度的预测性能均优于逻辑回归。Koroglu等人使用了七个旧版本的软件及其相关特征来检测当前版本的缺陷。他们比较了使用朴素贝叶斯，决策树和随机森林建立的模型。通过将比对这些模型的AUC值，他们发现随机森林具有最高的AUC值，发现研究的几种模型中，随机森林具有最高的预测能力。文献[11]提出一种名为ROCUS 的半监督学习方法，该方法结合了基于分歧的半监督学习和欠采样策略，尝试同时解决数据标记样本数量少以及数据不平衡问题，在8个实际软件项目上对所提方法进行了实证研究。薛参观等提出一种改进深度森林算法—深度堆叠森林（deep stack forest，DSF）。该算法首先采用随机抽样的方式对软件的原始特征进行变换以增强其特征表达能力，然后用堆叠结构对变换特征做逐层表征学习，采用正确率、准确率、召回率和F-1度量4个评价指标对实验结果进行评估，比较了其所提方法与深度森林模型的预测性能。

## 4.2 基于回归的技术

最初，Graves等人提出了一种在软件模块中预测缺陷数量的方法。缺陷预测模型是使用广义线性回归模型和软件变更历史指标建立的。结果发现，缺陷数量的预测提供了更多有用的信息，而不是预测模块是否有缺陷和无缺陷。文献[14]探索了决策树回归（DTR）在两种不同情况下对给定软件系统的版本内预测和版本间预测的缺陷数量预测的能力。实验研究是在五个开源软件项目中进行的，这些项目的19个版本是从PROMISE数据存储库中收集的。使用绝对误差和相对误差，来评估DTR的预测精度。

## 4.3 基于聚类的技术

缺陷预测方法使用软件指标和缺陷数据来了解哪些软件属性与类中的缺陷相关联。文献[15]指出现有技术在主题软件系统的相同发行版或后续发行版中预测易于出错的类。提出了一种内部释放缺陷预测技术，该技术从相关类的集群中学习，而不是从整个系统中学习。使用结构信息对类进行聚类，并使用识别出的每个类中对类的度量来构建缺陷预测模型。对来自PROMISE存储库的8个开源软件系统的9个发行版中的数据进行了实证研究，并使用多元线性回归构建了预测变量。结果表明，基于聚类的预测模型优于基于系统所有类的预测模型。文献[16]研究了混合模糊遗传算法和基于模糊聚类的技术。

综上所述，研究者采用机器学习技术在软件缺陷预测领域进行了大量的探索，但我们通过研究有以下发现：

（1）对于软件缺陷预测的相关问题研究，更多的研究集中于对软件模块的缺陷倾向性进行描述，而对于软件模块的缺陷数和缺陷严重性的研究相对较少。

（2）采用机器学习建立的模型大多采用默认参数，而对参数改变对模型性能的影响研究相对较少。

（3）机器学习技术具有可接受的预测能力，其AUC的均值介于0.7~0.83之间，精确度介于75%~85%之间。

（4）所有的研究建立的模型的数据依赖性较强，即针对特定的数据集有较好的预测性能的模型在其他数据集上并不能取得预想的结果。

# 第五章 结语

软件缺陷预测可以帮助单位或个人在软件项目管理中较早的发现项目中的漏洞，合理分配资源，提高生产效率，降低生产成本，提高软件质量，因此，近年来该领域的研究获得了大量的关注。本文从缺陷预测的概念、常用数据集、缺陷度量元和缺陷模型的构建等几个方面进行了简要的分析，可为相关研究提供参考。

# 参考文献

[1] 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.

[2] 宫丽娜,姜淑娟,姜丽.软件缺陷预测技术研究进展[J].软件学报, 2019, 30(10): 3090-3114.

[3] 吴方君.静态软件缺陷预测研究进展[J].计算机科学与探索, 2019, 13(10): 1621-1637.

[4] Naik K,Tripathy P. Software Testing and Quality Assurance || System Test Planning and Automation[M]. [S.l.]: [s.n.], 2008: 355-407.

[5] Mcdonald M,Musson R,Smith R, et al. The Practical Guide to Defect Prevention[J]. Pearson Schweiz Ag, 2007.

[6] Chidamber S,Kemerer C. A Metrics Suite for Object Oriented Design[J]. Software Engineering Ieee Transactions on, 1994, 20(6): 0-493.

[7] Abreu FBE, Carapu? a R. Candidate Metrics for Objectoriented Software Within a Taxonomy Framework[J]. Journal of Systems & Software, 1994, 26(1): 87-96.

[8] Madeyski L,Jureczko M. Which Process Metrics Can Significantly Improve Defect Prediction Models? an Empirical Study[J]. Software Quality Journal, 2015, 23(3): 393-422.

[9] Singh Y,Kaur A,Malhotra R. Empirical Validation of Object-oriented Metrics for Predicting Fault Proneness Models [J]. Software Quality Journal, 2009, 18(1): 3.

[10] Koroglu Y,Sen A,Kutluay D, et al. Defect Prediction on a Legacy Industrial Software: a Case Study on Software with Few Defects[C]//2016 Ieee/acm 4th International Workshop on Conducting Empirical Studies in Industry (cesi), [S.l.]: [s.n.], 2016: 14-20.

[11] 姜远,黎铭,周志华. Software Defect Detection with Rocus[J]. Journal ofComputerScience&Technology, 2011, 26(2): 328-342.

[12] 薛参观,燕雪峰. 基于改进深度森林算法的软件缺陷预测 [J]. 计算机科学, 2018, 45(8): 160-165.

[13] Graves TL,Karr AF,Marron JS, et al. Predicting Fault Incidence Using Software Change History[J]. Ieee Transactions on Software Engineering, 2000, 26(7): 653-661.

[14] Rathore S,Kumar S. A Decision Tree Regression Based Approach for the Number of Software Faults Prediction [J]. Acm Sigsoft Software Engineering Notes, 2016, 41(1): 1-6.

[15] Scanniello G,Gravino C,Marcus A, et al. Class Level Fault Prediction Using Software Clustering [C]//2013 28th Ieee/ acm International Conference on Automated Software Engineering (ase), [S.l.]: [s.n.], 11: 640-645.

[16] Gupta K,Kang S. Fuzzy Clustering Based Approach for Pre-diction of Level of Severity of Faults in Software Systems [J]. International Journal of Computer and Electrical Engineering, 2011, 845-84