

基于三层知识模型的软件缺陷管理系统

柴海燕, 丁霞, 王凯风, 谢立鹏

(京东方科技集团股份有限公司, 北京 100176)

摘要: 目前的缺陷分析还是基于数据层面分析与预测, 对软件测试工作的价值贡献受到局限; 文章设计了一种基于三层知识模型的软件缺陷管理系统; 通过把知识库的结构引入到软件缺陷管理中, 把缺陷分析从数据管理层面提升为知识管理层面; 通过事实知识层、中间层、策略层三层知识库结构对软件开发测试过程中产品产生的技术数据、管理过程数据进行采集沉淀, 提取软件缺陷预测规则, 并生成能指导软件测试工作的策略报告; 有助于从根本上提升软件测试工作水平和效率。

关键词: 知识库; 数据挖掘; 缺陷管理; 软件测试

Software Defect Management System Based on Three-layer Knowledge Model

Chai Haiyan, Ding Xia, Wang Kaifeng, Xie Lipeng

(BOE Technology Group Co., Ltd., Beijing 100176, China)

Abstract: Current defect analysis is based on data level analysis and prediction, and the value contribution to software testing work is limited. This paper designs a software defect management system based on three-layer knowledge model. By introducing the structure of the knowledge base into software defect management, the defect analysis is upgraded from the data management level to the knowledge management level. Through the three-layer knowledge base structure of fact knowledge layer, middle layer and strategy layer, the technical data and management process data generated by the software during the software development test process are collected and precipitated, the software defect prediction rules are extracted, and a strategy report capable of guiding the software test work is generated. It helps to fundamentally improve the level and efficiency of software testing.

Keywords: data mining; defect management; software testing

0 引言

通过在软件上线前对软件进行测试发现软件缺陷并进行修正来提升软件质量, 软件测试直接关系到软件质量。软件测试研究的热点领域有测试用例自动生成、测试用例约简、回归测试、缺陷分析等^[1-4]。测试用例的约简是一个 NP (non-deterministic polynomial) 问题, 各种算法基于各种均衡搜索最重要的测试用例, 然后用减少测试用例数量的方法提升软件测试效率。这在理论上有一定价值, 但在实际工作中并不接受通过牺牲缺陷检出率而换来的工作时间的减少, 因为任何一个软件缺陷都会对软件所在的系统造成致命的影响。缺陷分析研究领域基于历史缺陷数据进行分析并生成测试报告指导测试工作, 比单纯减少测试用例数量更有价值。但目前的缺陷分析还是基于数据层面分析与预测, 对软件测试工作的价值贡献受到局限。通过把知识库的结构引入到软件缺陷管理中, 把缺陷分析从数据管理层面提升为知识管理层面, 通过机器学习人工智能的方式提取规则知识, 并生成能指导软件测试工作的策略报告, 有助于从根本上提升软件测试工作水平和效率。

1 软件缺陷分析

在 ISO 9000 中对软件缺陷进行了定义, 软件缺陷是指

收稿日期: 2019-04-24; 修回日期: 2019-07-05。

作者简介: 柴海燕(1984-), 女, 陕西人, 工程师, 主要从事物联网软件测试方向的研究。

“未满足与预期或者规定用途有关的要求”。无论开发人员经验多么丰富都不能保证所设计的软件中没有缺陷。软件缺陷伴随着软件的整个生命周期, 当触发条件满足时就会触发缺陷造成难以估量的损失。软件测试工作就是在软件上线之前发现软件中缺陷。

软件缺陷分析通过对软件开发测试过程产生各种缺陷进行追踪分析。借助电子化辅助的方法对软件缺陷进行记录、分类整理、统计分析, 可以提升这个缺陷分析的效率^[5-10]。但目前缺陷分析仅仅限于对缺陷进行统计度量。缺陷的分类方法有很多, IEEE 组织提出了的软件异常分类法, IBM 公司提出了正交缺陷分类法, 我国军方相应标准中也对软件缺陷进行了定义分类, 例如《GJB 2786A 军用软件开发通用要求》中关于软件错误的分类。

2 知识库与机器学习

知识库是知识型人工智能的核心。知识库中的知识分为三层: 事实知识层、中间层、策略层。其中中间层是基于事实知识的规则过程, 策略层是基于中间层知识的规则。

人工智能不能只重视过程方法, 说明性方法与经验同样重要。缺陷分析有很多相关研究, 这些研究大多从算法优化角度优化缺陷分析工作, 这种研究方向过于局限。把知识库引入到软件缺陷管理中, 可以把缺陷分析管理从缺陷数据管理提升对软件开发测试全流程知识管理。

机器学习是一门人工智能的科学, 基于数据和算法学习经验提升算法性能。机器学习采用的学习策略分为 4 种:

机器学习、传授学习、类比学习、通过示例学习。通过知识库、机器学习算法把传统的缺陷管理系统升级为软件测试专家系统。

3 缺陷管理系统功能结构

在基于知识库数据挖掘的软件缺陷管理系统中通过知识库架构对缺陷进行管理,用三层知识架构把缺陷管理从数据层面提升到知识层面。本文设计的缺陷管理系统从 4 个方面进行软件测试缺陷管理,缺陷数据管理、缺陷规则知识挖掘、缺陷预测、缺陷策略知识配置。其中缺陷数据管理对应知识库的事实知识层,缺陷规则知识挖掘对应知识库的中间层,缺陷预测与策略配置对应知识库的策略知识层。

事实知识层是知识库的基础层,事实知识层的知识来源于各周边交互系统或相关人员。事实知识层所采集的数据贯穿软件开发测试工作的全流程,事实知识层所采集的数据质量和数量决定这系统的可用性和有效性。

软件开发是一项系统工程,造成软件缺陷的因素很多,涉及软件开发工程的每个环节。从软件本身角度看,软件功能复杂度、代码量、软件架构等因素与软件缺陷都相关。从软件开发过程来看,开发过程中的需求沟通、文档质量、变更过程版本记录都会影响软件中缺陷的多少。从开发项目管理角度来看,项目时间计划安排不合理、PM (project manager) 人员过程管理不科学、软件开发人员的经验缺乏、开发人员流动大都可能引发软件缺陷。从软件开发方公司层面角度来看,软件开发公司规模、公司资质、项目是够外包等都与软件质量相关。对于第三方测试公司来讲,无法控制软件开发过程,但尽可能多的获取软件开发相关信息数据了解软件开发情况对软件测试工作很有意义。可以通过客户关系管理系统对相关数据进行收集管理。

同样软件测试也是一项系统工程,软件开发工程决定了软件缺陷的数量。软件测试工作决定了检测出的软件缺陷数量。未检测出的软件缺陷会造成质量问题,带来各种隐患。从软件测试过程来看,测试人员、测试文档、测试用例设计、缺陷历史数据的利用等因素都会与检测出的缺陷数量相关同样直接关乎软件质量。

软件开发与软件测试过程中积累了很多与软件缺陷相关的数据,有效管理利用这些海量数据能提升软件测试工作的效率。软件缺陷管理要贯穿整个软件开发与软件测试过程,所以缺陷数据采集也需要贯穿软件开发与测试全流程。

事实知识层面通过查询接口已经能支撑测试人员分享获取软件测试相关知识。但事实知识层数据量很大,只适合查询形式的场景。基于事实知识层通过机器学习算法进行数据挖掘形成规则知识层的规则知识库。事实知识层面的知识只是在描述事实,而规则层知识则是对事实知识的智能抽取和提炼形成的经验规则。

基于规则知识库中的知识对软件进行预测并配置相应的策略就是形成了对软件测试工作有价值的策略层知识。

知识库三层的知识对测试人员价值逐层递增,事实知

识层只能给相关人员提供事实知识的描述。规则知识层能提供通过机器学习算法在事实知识的基础上总结出的规则经验。而策略知识层可以对一个待测的软件项目进行缺陷预测并提供相应的测试策略报告来给测试相关人员提出测试相关的工作建议。

例如,事实知识记录了软件开发公司、开发人员、模块分支数、模块代码行数、缺陷信息等数据知识,通过机器学习算法在这些数据的基础上进行抽取可以得到一个规则知识,“A 公司张某某开发的代码行数超过 900 行分支路径大于 10 的程序模块会大概率出现缺陷”。通过策略知识层可以对一个待测的软件项目输出缺陷预测报告以及测试用例设计指导报告。

基于知识库数据挖掘的软件缺陷管理系统核心功能实现包括 8 个模块:事实知识数据库、数据预处理模块、机器学习算法模块、规则抽取模块、规则知识库、缺陷预测模块、策略知识库、账号权限管理与查询交互子系统。

事实知识数据库是缺陷预测管理的基础,事实知识数据库存储着软件测试工作中产生的与软件缺陷相关的数据。事实知识数据仓库中的数据来源于几个方面:测试人员、业务人员、管理人员的输入以及从相关系统进行采集。事实知识数据包括:缺陷历史数据、项目管理数据、客户关系管理数据等几个方面的数据。缺陷历史数据来源缺陷追踪系统,项目管理数据来源于测试管理人员的录入以及开发方输入。客户数据来源于客户关系管理系统。在专业软件测试公司会有很多客户,相关客户关系管理系统中记录着客户相关信息。

数据预处理模块基于事实知识数据库中的数据进行 ETL (extract—transform—load) 处理形成基础事实数据表。基础事实数据表有两个类型的字段组成,缺陷标识字段以及测试属性字段。缺陷标识字段标识缺陷类型,测试属性数据是与测试用例、测试缺陷相关的维度属性。

机器学习模块把预处理模块输出的基础事实数据表作为输入,经过机器学习算法输出规则树。规则抽取模块根据缺陷管理目标抽取缺陷规则,形成规则知识。规则知识库存储规则知识。软件预测模块基于规则知识库中的规则对软件进行预测,策略知识库对预测结果配置相应的缺陷管理策略。

4 缺陷管理主要流程

4.1 基于知识共享的测试交互流程

缺陷管理系统基于知识库进行设计,知识库本质是进行知识的加工与分享。这种共享不仅是测试人员之间、测试人员与项目管理人员,而且是测试人员、测试项目管理人员与软件开发方的知识共享。所以基于此系统进行缺陷管理其实是一个知识共享的过程。基于知识共享的测试流程涉及的实体包括:软件开发方、缺陷管理系统、项目管理人员、测试人员。

开发方提供相关数据到缺陷管理系统。相关管理人员

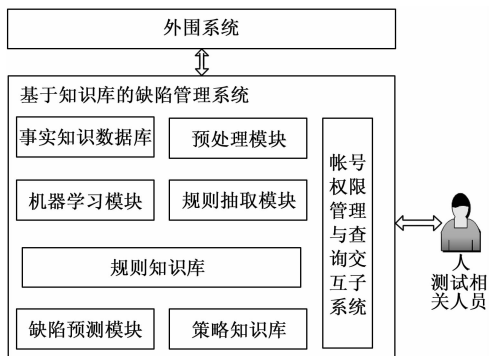


图 1 系统功能模块图

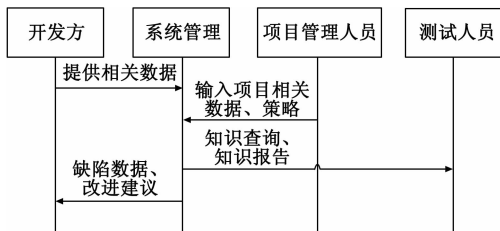


图 2 基于知识共享的测试交互流程图

向缺陷管理系统输入项目相关数据、配置相应策略。管理系统给测试人员提供知识查询、知识报告服务。管理系统给开发方提供缺陷相关信息以及改进建议，将测试结果以及缺陷建议报告反馈给开发方。

4.2 规则知识挖掘流程

从事实知识到规则知识需要经过机器学习算法进行数据挖掘。数据挖掘的流程包括数据预处理、机器学习、规则抽取、规则评估、规则入库等 5 个步骤。

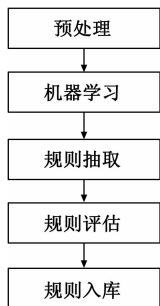


图 3 规则知识挖掘流程

第一步：数据预处理就是把事实知识数据库的数据进行加工处理成机器学习算法所要求的数据输入格式。事实知识数据库的数据包括测试历史数据、缺陷历史数据 (Software Defect Management Data)、项目管理数据 (project management data, PMD)、客户关系管理数据 (customer relationship management data, CRMD)。测试历史数据主要包括测试用例、测试人员，缺陷历史数据记录着软件测试工作测试出的缺陷相关信息，包括被测软件项目信息、测试人员信息、测试用例数据、缺陷相关数据。项目管理数据包括项目描述数据、开发人员数据、开发过程数

据、需求文档、设计规范文档、源代码数据。客户关系管理数据，软件开发方公司相关数据。这些数据处理成由缺陷标识字段与属性字段组成的数据表。

数据表按照 8 : 2 的比例把数据表记录分为训练数据集合和验证数据集合。训练数据集合用于训练生成规则，验证数据集合用于验证规则的性能。

第二步：基于前面步骤生成的缺陷标识、属性字段训练数据表进行数据挖掘形成规则树。数据挖掘的过程由机器学习算法完成。缺陷历史数据中包括软件缺陷情况以及缺陷类型，所以可以用有监督机器学习模型进行数据训练，找到缺陷类别与属性数据之间的规则。缺陷预测问题可以抽象成分类问题。这里采用决策树算法对数据进行训练学习生成规则树。

第三步：在规则树的基础上进行规则抽取。规则树每个根节点以及中间节点对应个属性，叶子节点代表缺陷类别。每一条从规则树根节点到叶子节点的路径抽取成一条规则。遍历每条从根节点到叶子节点的数据路径就完成了规则抽取过程。

第四步：对抽取的规则在验证数据集合上进行验证，得到精准度、覆盖度等指标。根据设定的指标阈值对指标进行取舍，保留符合指标要求的规则。

第五步：前面符合要求规则入到知识库的规则库，入库后这些规则就形成了规则知识层中的知识。

4.3 缺陷预测与策略报告生成流程

事实规则知识经过规则知识挖掘流程的处理得到规则知识，基于这些规则知识可以进行缺陷的预测输出缺陷预测报告。缺陷预测不是最终目的，对预测到缺陷采取相应的应对措施才能发挥出规则知识的价值。缺陷预测与策略生成由策略知识层控制。策略知识层的输入为软件项目资料数据，输出为缺陷预测报告以及策略生成报告。

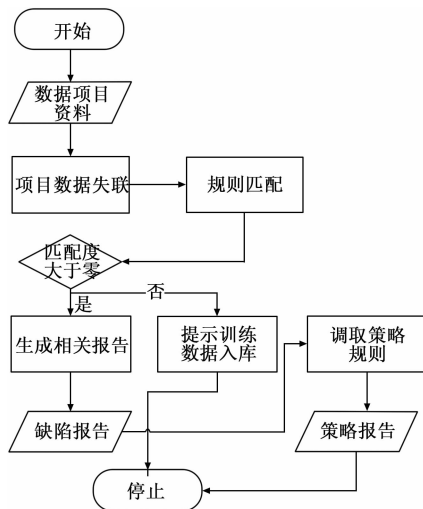


图 4 缺陷预测与策略报告生成流程图

缺陷预测与策略报告生成流程主要包括项目数据关联、
(下转第 140 页)