

# 基于神经网络的软件质量评价综述

宗鹏洋 王轶辰

(北京航空航天大学可靠性与系统工程学院 北京 100083)

**摘要** 软件质量贯穿于软件生存周期的一个重要问题,随着软件产业的发展,人们对软件质量的要求也越来越高,因此如何建立准确客观的软件质量评价模型成为软件质量领域研究的重要课题。软件质量评价模型旨在从历史数据中寻找软件各个方面的特征与软件质量之间的关系,而神经网络因其强大的学习能力与非线性映射能力成为建立这种复杂关系的最合适的方法。为了总结现有的相关研究并为以后的研究提供思路,以系统性文献综述的方法调研了自 1994 年至 2018 年国内外 50 篇使用神经网络方法进行软件质量评价的文献,从输入元素、评价目标、建模方法以及神经网络的训练等方面对文献进行了归纳与总结,发现了使用神经网络方法进行软件质量评价的一些规律、未解决的问题以及可能的研究方向。

**关键词** 软件质量,质量评价,神经网络,系统性文献综述

**中图分类号** TP311.5 **文献标识码** A

## Software Quality Evaluation Based on Neural Network: A Systematic Literature Review

ZONG Peng-yang WANG Yi-chen

(Science & Technology on Reliability & Environment Engineering Laboratory, Beihang University, Beijing 100083, China)

**Abstract** Software quality is a significant factor throughout the software life cycle. With the rapid development of software industry, users have higher and higher requirements on software quality. Therefore, how to establish a more accurate software quality evaluation model has become an hot topic in the field of software quality research. The software quality evaluation model aims to find the relationship between the characteristics of various aspects of software and software quality from historical data. And neural network becomes one of the most appropriate methods to establish such a complex relationship because of its powerful learning ability and non-linear mapping ability. Using the method of systematic literature review, this paper summarized 50 domestic and foreign literatures on software quality evaluation using neural network method from 1994 to 2018 from the aspects of inputs, evaluation targets, modeling methods and the training of neural network. Some rules, unsolved problems and possible research directions of using neural network method to evaluate software quality were found.

**Keywords** Software quality, Quality evaluation, Neural network, Systematic literature review

## 1 引言

软件质量是软件的生命,软件质量评价是保证软件质量的重要手段<sup>[1]</sup>。由于软件质量是一个抽象而又复杂的集合,因此建立准确客观的软件质量评价模型是一项复杂的系统工程,也是软件质量领域的重要课题。

随着神经网络技术的迅速发展,神经网络技术已成为构造经验模型的首选方法<sup>[2]</sup>,其强大的非线性映射能力与自学能力很适合作用来从历史数据中寻找软件各个方面的特征与软件质量之间的关系,从而建立软件质量评价模型。

系统性综述是针对明确定义的特定问题,针对其既有文献对其使用精密、可重复性、造成最低偏倚的方法进行高度严密的评价,系统地搜索、分辨、选择、评价和综合与课题有关的研究证据<sup>[3]</sup>。系统综述本身也是一项研究,它能够解决比单一研究更广泛的问题,如揭示许多研究之间的联系。

本文旨在对软件质量评价中的神经网络方法进行系统综

述,归纳总结了评价模型的输入元素、评价目标、建模方法与神经网络训练等方面的内容,并发现了使用神经网络方法进行软件质量评价的一些规律、未解决的问题以及可能的研究方向。

## 2 调研方法

《在软件工程领域中应用系统文献综述的经验教训》<sup>[4]</sup>描述了如何在软件工程领域进行系统性文献综述,本综述的调研按照其指导分为计划、执行和报告 3 步进行(见图 1)。

**第 1 步:计划阶段。**在这一阶段,我们确定了调研范围与研究问题,并以此制定了调研策略。调研策略包括文献检索的条目与数据库、文献的筛选规则、质量评估的规则、数据提取与分析的方法等。最后,我们选择了一个数据库对调研策略进行初步的验证,针对实际的情况做出了调整。

**第 2 步:执行阶段。**在这一阶段执行第一步制定的调研策略。对选定的数据库按照特定的关键字进行检索,并按

筛选规则将检索结果纳入或排除到我们的研究中。对最终确定的研究文献进行质量评估,并提取所需数据进行统计分析。

第3步:报告阶段。在这一阶段撰写调研报告,也就是本篇文献综述的结果与讨论部分。

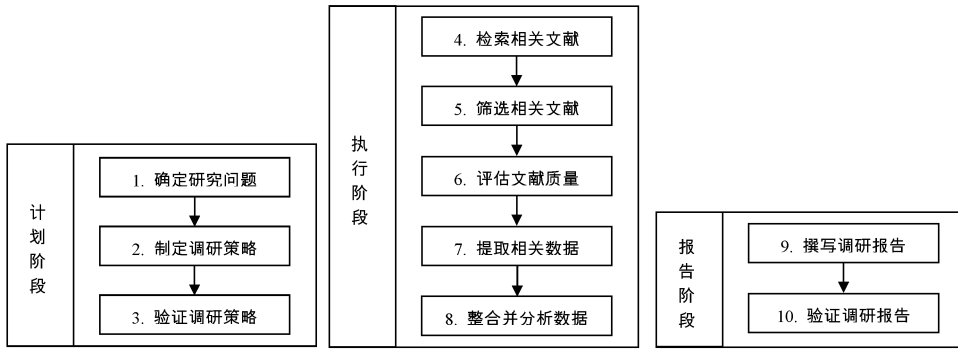


图1 调研步骤

本节主要介绍计划与执行阶段的细节,因此将在以下4个小节中分别介绍研究问题的确定、检索策略的确定以及检索过程与结果、筛选规则的确定以及筛选结果、数据提取与整合的方法。

## 2.1 研究问题

本文的目的在于总结应用神经网络方法进行软件质量评

价的文献,因此提出的研究问题也主要针对软件质量评价模型及其神经网络方法两方面展开。对于文献提出的软件质量评价模型,我们关注其输入元素(RQ1)、评价目标(RQ2)以及使用的方法(RQ3);对于神经网络方法,我们关注文献给出的训练过程、验证方式以及可能的优化方法(RQ4)。表1列出了我们提出的研究问题以及提出此问题的具体动机。

表1 研究问题与动机

问题	动机
RQ1 软件质量评价模型的输入是哪些元素?	调查影响软件质量而且方便获取的软件因素
RQ1.1 输入元素为什么可以用来评价软件质量?	探究输入元素用于评价软件质量的合理性
RQ1.2 输入元素是否受开发方法或编程语言的限制?	调查模型的应用是否受限制
RQ1.3 输入数据的来源是什么?	数据来源可以反映模型的可信性
RQ1.4 有哪些工具支持输入数据的采集?	调查有哪些商用或开源的数据采集工具用于软件质量评价模型的自动化数据采集
RQ2 软件质量评价模型评价的是什么指标?	调查可以直接反映软件质量的指标
RQ2.1 软件质量的这些指标为什么可以代表软件质量?	探究模型精简处理的理由
RQ2.2 哪些软件质量模型为其提供了评价目标?	调查软件质量模型在软件质量评价中的应用
RQ2.3 评价目标数据是主观的还是客观的?	探究模型的客观程度,是模型可信性的体现
RQ2.4 评价目标数据的来源是什么?	数据来源也可以反映模型的可信性
RQ3 有哪些方法用于软件质量评价?	调查软件质量评价建模常用的算法
RQ3.1 神经网络方法的优势是什么?	探究采用神经网络方法建模的原因
RQ3.2 采用了哪种神经网络模型?	调查用于建模的神经网络种类及其特点
RQ3.3 哪些方法可以用于神经网络的优化?	调查可以增强神经网络性能的方法
RQ3.4 除了神经网络还提供了哪些方法作为对比?	调查其他用于评价模型的建模方法
RQ4 神经网络是如何训练的?	研究文献在实证中训练的神经网络
RQ4.1 训练数据是否需要预处理?	探究数据预处理的必要性及其方式
RQ4.2 训练集和测试集或验证集是如何分配的?	调查数据按什么比例划分,测试集或验证集的存在也是模型可信性的体现
RQ4.3 神经网络的参数是如何设置的?	调查神经网络模型的常用参数配置
RQ4.4 哪些指标用于验证神经网络的性能?	调查用于验证神经网络性能的常用指标

## 2.2 检索策略

检索策略包括检索的数据库与检索关键字的制定。我们从研究可用数据库中选取了4个涵盖了工程技术或工业技术类文献的中文数据库,以及6个涵盖了计算机科学领域的外文数据库,数据库列表如表2所列。

表2 检索数据库列表

中文数据库	外文数据库
超星期刊	ACM Digital Library
万方数据	ScienceDirect
维普期刊	EV2
中国知网	IEEE Xplore
	ProQuest
	Web of Science

没有限制文献对软件质量进行评价、评估、预测或是其他行为。我们计划在文献标题中检索“软件”与“质量”来将文献限制在软件质量领域,在标题或关键词中检索“神经网络”或“深度学习”(英文检索关键字考虑了同义词“Neural Network”、“NN”、“Deep Learning”)以检索使用神经网络方法的文献,通过布尔运算符 AND 与 OR 将它们连接为检索式(Search strings)进行检索。

由于此前没有学者对软件质量评价中的神经网络方法进行系统的总结,因此我们检索了从1994年到2018年的相关文献,检索结果如表3所列。检索结果中包含大量重复文献,我们去除重复文献后得到的文献数量为中文文献45篇,英文文献101篇。

为了避免遗漏相关文献,我们使用了粗略的检索关键字,

表 3 检索结果

中文数据库	结果/篇	英文数据库	结果/篇
超星期刊	32	ACM Digital Library	27
万方数据	47	ScienceDirect	8
维普期刊	29	EV2	65
中国知网	39	IEEE Xplore	22
		ProQuest	40
		Web of Science	25

### 2.3 筛选规则

我们在检索阶段制定了粗略的检索关键字来避免遗漏文献,也就意味着同时会有许多无关文献被检索到,这就要求我们制定严格的筛选规则,纳入相关的文献并排除无关文献。筛选规则中的纳入和排除规则是客观而且始终一致的,具体纳入规则如下:

纳入规则 1 使用神经网络方法进行软件质量评价(或评估)的论文;

纳入规则 2 使用神经网络方法进行软件质量预测的论文;

纳入规则 3 将神经网络方法与其他软件质量评价(或评估)建模方法比对的论文;

纳入规则 4 将神经网络方法与其他软件质量预测建模方法比对的论文。

排除规则如下:

排除规则 1 没用使用神经网络方法建模的论文;

排除规则 2 使用神经网络方法但与软件质量无关的论文;

排除规则 3 研究软件生产线质量、软件需求质量与数据质量的论文;

排除规则 4 会议论文在期刊有 Extended Version 的,排除旧版论文。

最终我们根据筛选规则纳入了 50 篇相关文献,包括 15 篇中文文献与 35 篇英文文献。我们将中文文献以 CS (Chinese study) 开头编号,英文文献以 ES (English study) 开头编号。

### 2.4 数据提取与分析

提取并整合数据的目的是从选择的文献中收集事实与数据并将它们联系起来,从而解答相应的研究问题<sup>[5]</sup>。通过分析和比较相似或者相悖的事实与观点,我们可以得到研究问题的答案。因此,我们按照需要调研的研究问题提取了 50 篇文献的 5 类数据,分别为文献的元数据(Metadata)、输入数据、输出数据、建模数据与神经网络数据。进一步根据研究问题的子问题将 5 类数据细分为 30 类子数据,具体类别如表 4 所列。我们将提取出的 30 类子数据都储存在 Excel 表格中,以便得到可视化的分析结果。

表 4 数据类别

数据类别	数据子类别
文献元数据	题目、作者、出版时间
输入数据	输入元素、语言限制、输入数据来源、输入获取工具
输出数据	输出元素、参考软件质量模型、输出主客观、输出数据来源
建模数据	神经网络类型、采用对比技术、神经网络优化方法、数据预处理方法
神经网络数据	输入层单元数、隐层单元数、模糊条件层单元数、模糊规则层单元数、输出层单元数、激活函数、学习率、动量、代价函数、停止训练条件、设置迭代次数、实际迭代次数、训练集大小、测试集大小、性能验证指标

## 3 结果与讨论

本节将展示对初步筛选的 50 篇文献的分析结果,首先对文献的元数据(Metadata)进行分析,之后是对每个研究问题的详细解答,并对研究问题得到的答案进行解释与讨论。

### 3.1 元数据

在元数据中,我们选择了文献的作者与出版时间进行结果展示,出版源也是我们分析的数据之一,但没有得到有价值的信息。由于文献的出版年份跨度较大,我们将 1994 年到 2018 年分成了 7 个时间段,以研究论文按发表时间的分布。从图 2 可以看出,对神经网络在软件质量评价中的应用的论文数量在 2003 年之后有小幅度的上涨,但论文数量总体比较稳定,说明此技术从开始至今一直收到持续的关注。也可以看出,神经网络方法一开始的应用局限于对软件单方面质量因素的评价,后来逐步应用到软件质量综合评价中。

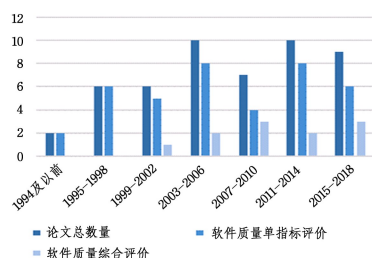


图 2 文献数量

在我们纳入的 50 篇文献的作者中,多次出现的作者如表 5 所列,而表 6 列出了引用量最多的 3 篇文献(引用量数据来自谷歌学术)。

表 5 多次出现的作者列表

姓名	次数	论文编号
Khoshgoftaar T M	5	ES1、ES2、ES3、ES4、ES10
Szabo R M	2	ES2、ES3
Pedrycz W	3	ES8、ES9、ES12
Peters J F	3	ES8、ES9、ES11
Ramanna S	2	ES8、ES11
Yu B	2	ES16、ES18
Wang Q	2	ES16、ES18
Yao L	2	ES19、ES20
Pomorova O	2	ES22、ES27
Hovorushchenko T	2	ES22、ES27
Mishra R	2	ES30、ES34
Sheoran K	2	ES30、ES34
Tomar P	2	ES30、ES34
官丽娜	2	CS11、CS14
马怀志	2	CS11、CS14

表 6 高引用量论文列表

编号	作者	时间	被引量
ES4	Khoshgoftaar T M, Allen E B, Hudepohl J P, et al	1997	225
ES14	Mie Mie Thet Thwin, Tong-Seng Quah	2003	204
ES15	Taghi K, Naeem S	2003	148

### 3.2 研究问题结果

RQ1 软件质量评价模型的输入是哪些元素?

使用神经网络方法的评价模型都是多输入的,这些输入元素往往来自于作者挑选的某个特定集合,这个问题主要对这些特定的集合以及集合内包含的输入元素进行调研。我们将这些特定集合分为两类,第一类为软件度量,包括软件代码

度量(35次)、项目度量(4次)与测试度量(1次),第二类为软件质量特性(5次)或其子特性(5次)。

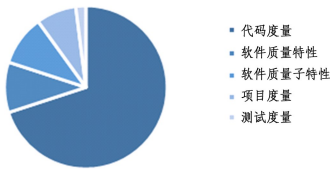


图3 输入元素

软件的代码度量是指可以直接从代码中得到的度量。使用代码度量作为评价模型,为输入的35篇文献提供了丰富的代码度量种类,除了McCabe结构复杂性度量、Halstead文本复杂性度量、Henry&Kafura信息流度量、C&K面向对象度量、MOOD面向对象度量等已被广泛应用的软件度量集合,还有许多作者提出了自己的软件度量组合。

图4统计了出现频率大于等于5次的软件代码度量。圈复杂度是基于神经网络的评价模型使用最多的软件度量,有40%的代码度量组合中包含了圈复杂度(ES1, ES2, ES3, ES4, ES5, ES7, ES9, ES10, ES11, ES22, ES24, ES25, CS1, CS10)。而使用最多的度量集合是C&K面向对象度量集(ES14, ES17, ES19, ES20, ES21, ES28, ES29, ES33, ES35, CS3, CS11, CS15),包括类中方法权重(WMC)、继承树深度(DIT)、直接子类数(NOC)、对象间的耦合(CBO)、响应集大小(RFC)与方法内聚缺乏度(LCOM)。

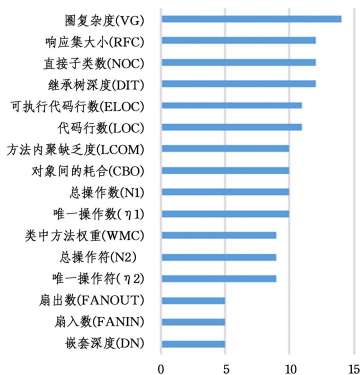


图4 软件代码度量

#### RQ1.1 输入元素为什么可以用来评价软件质量?

正如上述分析,输入元素主要分为软件度量与软件质量特性或其子特性两种。软件质量模型提供了软件质量特性及其子特性与软件质量的直接关系,但是使用软件度量来评价软件质量并没有直接的理论依据支撑,其合理性需要进一步探究。

一些文献基于软件度量的定义给出了这种合理性。IEEE“Standard for Software Quality Metrics Methodology, IEEE Std, 1061-1992, 1993”中定义度量为一个函数,输入软件数据来输出能用来解释软件所具有的一个给定属性对软件质量影响的程度的单一数值。CS3指出软件质量度量就是指对影响软件质量的属性所进行的定量的测量与度量的方法。有的学者也给出了类似的定义,软件度量就是对影响软件质量的内部属性进行量化测量。而软件的可靠性、功能性等这些质量要素是无法直接度量的,需要通过度量软件的内部属性来间接地得到(CS14)。

也有一些文献从实践方面考虑,使用软件度量的原因是

方便调整开发过程。ES3指出通常我们希望使用一些可量化的度量来预测软件系统的质量,并基于这种预测来调整开发过程以提高软件的整体质量。但是ES13提出了质疑,他们认为使用诸如操作数和操作符的数量之类的软件度量可以预测软件质量,但并不能真实地反映软件质量。

#### RQ1.2 输入元素是否受开发方法或编程语言的限制?

这个研究问题主要考虑软件质量评价模型的适用情况,如使用面向对象度量作为输入的模型无法适用于面向过程语言开发的软件中,同样对于面向对象语言开发的软件,一些非面向对象类的复杂性度量可能不足以描述软件的内部特征。因此我们调研了文献中案例部分采用的软件所使用的开发语言,结果如图5—图7所示。有11篇文献的评价模型针对面向对象语言设计的软件进行建模,包括C语言(ES1, ES7, ES16, ES18),汇编语言(ES2, ES3, ES9),PASCAL(ES5)与类PASCAL语言PROTEL(ES4, ES15)等。有15篇文献的评价模型针对面向对象语言设计的软件进行建模,包括C++(ES17, ES21, ES23, ES24, ES26),Java(ES12, ES34)和Ada(ES35)等。

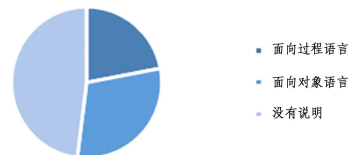


图5 数据来源语言类型

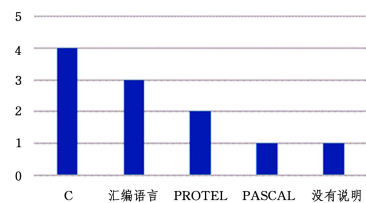


图6 面向过程语言

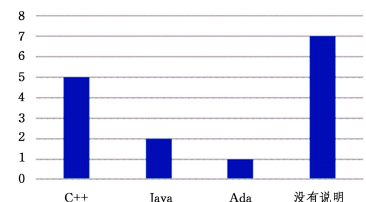


图7 面向对象语言

#### RQ1.3 输入数据的来源是什么?

输入数据是基于神经网络的软件质量评价模型训练数据的一部分,其来源反映了训练出的模型的可信程度。在10篇使用软件质量特性与子特性作为输入的评价模型中,其中7篇的输入数据来源为专家打分(ES32, CS2, CS4, CS6, CS7, CS9, CS15),有1篇使用随机生成的数据(CS8)。使用软件度量作为输入的软件质量评价模型的输入数据来源如图8所示。电讯系统(Telecommunications System)为软件质量模型的建模提供了大量数据(ES1, ES4, ES6, ES7, ES10, ES15, ES16, ES18),自商业软件(ES2, ES3, ES14, ES33, CS13)、NASA公开数据集(ES23, ES24, ES26, CS14)、军用软件(ES13, CS1, CS9, CS10)、学生作业(ES5, ES17)或其他实际项目的数据也被收集用于评价模型的训练。ES28与ES35直接采用了其他文献提供的数据,ES31使用的项目度量的数据

来自模糊评分,而 ES3 是唯一的使用测试度量作为输入的评价模型,他们的数据来自工具自动生成的测试用例。

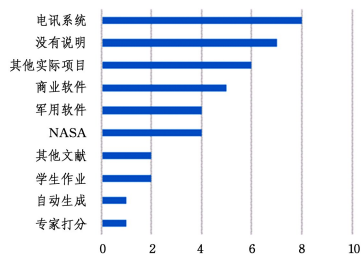


图 8 数据来源软件类型

#### RQ1.4 有哪些工具支持输入数据的采集?

此问题的目的是调研可以使用的自动化度量数据采集工具。ES3 采用自开发工具来采集所需软件度量,其他文献提及的软件度量数据采集工具有 DATRIX (ES4、ES16)、The Object-Oriented Metric Calculator (ES17)、Rational Rose (ES28)、Together (ES28) 与 CKJM (Chidamber and Kemerer Java Metrics) Tool (ES29)。

#### RQ2 软件质量评价模型评价的是什么指标?

软件质量模型将可以直接或间接反映软件质量的评价目标作为输出,我们对评价目标的统计展示在下面的图表中。软件综合质量是软件质量评价模型最关心的评价目标(14次),模型通过软件质量特性(CS4, CS6, CS8, CS9, CS12)、软件质量子特性(ES25, ES32, CS2, CS7, CS15)、项目度量(ES22, ES27, ES31)或代码度量(ES12)进行软件综合质量评价。软件的可靠性、效率、可维护性与可重用度等软件的某些质量特性也是软件质量评价模型关心的评级指标(ES8, ES19, ES20, ES28, ES30, ES33, ES34, CS11, CS13, CS14)。而软件缺陷、故障与失效方面的指标同样备受关注(ES1, ES3, ES4, ES5, ES7, ES10, ES13, ES14, ES15, ES16, ES17, ES18, ES21, ES23, ES24, ES26, ES29, CS1, CS3, CS5, CS10),反映软件可维护性的代码变更量(Change effort)也会通过软件度量来评价(ES2, ES6, ES9, ES11, ES14, ES35)。

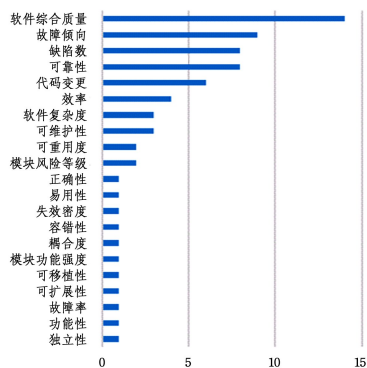


图 9 评价目标

#### RQ2.1 软件质量单个方面的指标为什么可以代表软件质量?

虽然所选文献建立的都是对软件质量的评价模型,但模型输出的指标各不相同。软件质量特性或其子特性这些评价目标虽然不是对软件质量的综合评价,但是对软件质量某个方面的描述。这个问题主要关心文献对于使用软件缺陷、故障、失效或代码变更量等指标代表软件质量的解释。

对于缺陷、故障或者失效等相关指标可以代表软件质量

的原因,以下文献给出了解释。CS3 的模型输出的是缺陷数,他们指出在软件测试开始阶段之前预测各个模块的潜在缺陷数、失效数,确定测试重点,合理分配测试的人力物力以及如何用度量来提高软件的质量一直是软件工程以期解决的问题之一。而关注模块风险等级的 ES1 表示通过在实施阶段识别出高风险模块,管理层可以将更多测试资源用于最有可能产生质量问题的模块。由于故障的高度倾斜分布(大多数组件没有故障或很少故障),ES5 建立的模型以故障倾向为输出,他们给出的理由是,在软件质量属性中我们关注的是故障倾向,因为大部分开发的时间和成本都花在了检测和修复故障上。

ES14, ES17 与 ES35 指出了故障、代码变更量与可靠性和可维护性之间的关系。在实践中,软件质量评估意味着要么评估可靠性,要么评估可维护性,模块的可靠性可以通过故障率来测量,而可维护性通常用代码变更量来度量(ES14, ES17)。可维护性取决于软件复杂度,而软件复杂度与每个类需要做出的变更数量成正比(ES35)。

ES28 所评价的是模块的可重用性,因为对于高质量软件系统的构建来说,可重用性起着重要的作用,软件重用可以通过减少设计和开发耗费的精力、时间和成本来提高软件的生产率和质量。

#### RQ2.2 哪些软件质量模型为其提供了评价目标?

软件质量模型常常为软件质量评价模型提供了参考的输入元素或评价目标。通过软件质量特性或其子特性对软件质量进行评价的模型往往基于软件质量分层模型来选择哪些质量特性与子特性作为模型的输入,有些通过软件度量来进行软件质量评价的模型也会选择软件质量特性作为评价目标。使用最多的软件质量分层模型是 ISO-9126 软件质量模型(ES19, ES20, ES32, CS4, CS6, CS8, CS9, CS15),其次是 McCall 软件质量模型(ES8, CS2),而 Boehm 软件质量模型与军事软件评估指标体系被分别用于 CS7 与 ES25 中。

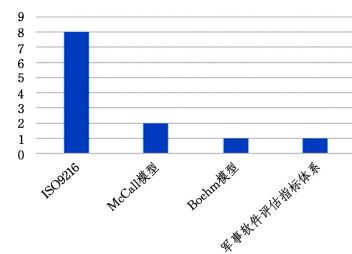


图 10 软件质量分层模型

#### RQ2.3 评价目标数据是主观的还是客观的?

基于神经网络的软件质量模型的训练采用的都是有监督学习算法,在训练模型时自然要求训练数据中除了输入元素的数据外还要包含标签数据(期望的输出)。这个问题我们关心文献训练模型所使用的训练数据中标签的主客观性,标签的主客观性会影响到模型的主客观性,也关系到模型的可信程度。来自软件测试、缺陷报告或者公开数据集的与软件失效、故障、缺陷或代码变更量相关的指标都是客观的标签,而对软件综合质量、软件质量特性的评价结果大多数文献采用专家评分来提供所需数据(ES12, ES19, ES20, ES25, ES31, ES32, ES33, CS2, CS4, CS6, CS7, CS8, CS9, CS13, CS15),难免会存在主观性。标签数据的具体来源将在下一个问题中展示。



图 11 标签数据的主客观性

## RQ2.4 评价目标数据的来源是什么？

正如 RQ2.3 中所描述的,对软件综合质量、软件质量特性的评价结果有 15 篇文章采用专家的主观评分来提供所需数据,也有文献采用公式计算了软件质量特性值(ES28, ES30, ES34)。专家打分可以分为专家对软件综合质量打分(ES12, ES25, ES31, ES32, CS2, CS4, CS6, CS7, CS8, CS9, CS15)与软件对可靠性,效率等软件质量特性打分(ES19, ES20, ES33, CS13)。与软件失效、故障、缺陷或代码变更量相关的指标来自于软件测试(ES4, ES5, ES6, ES9, ES13, ES16, ES17)、缺陷报告(ES1, ES7, ES15, ES18, CS1)、NASA 公开数据集(ES23, ES24, ES26)、代码控制或质量评估系统(ES2, ES3, ES14)等。

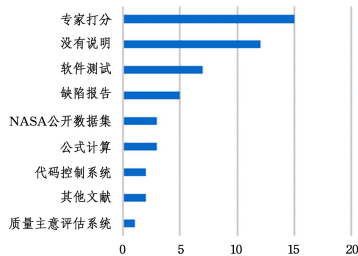


图 12 评价目标数据来源

## RQ3 有哪些方法用于软件质量评价？

## RQ3.1 神经网络方法的优势是什么？

大多数文献都给出了选择神经网络方法作为建模方法的原因,最主要的原因是神经网络的自组织、自适应与自学习的能力(ES6, ES20, ES31, ES32, ES33, CS1, CS7, CS12, CS15)。因为软件质量评价模型实际上是通过特定的数据处理方式从历史数据中寻找输入元素与软件质量评价目标之间的关系,是一种学习的过程,其次是神经网络的非线性映射能力(ES4, ES14, ES16, ES25, ES34, CS10),具体来说,神经网络擅长描述难以用其他技术描述的非线性函数关系,拥有通过学习逼近任何非线性映射的能力(ES14, ES25, ES34),而静态软件度量与软件质量特性之间的关系通常是复杂和非线性的,这限制了传统方法的准确性(ES4)。其他原因还有神经网络的联想记忆和分类功能(ES31, CS5, CS12)、鲁棒性(具体来说是神经网络处理噪声、损坏数据或不完整数据的能力)(ES13)、大规模并行处理能力(ES33, CS15)、高速寻找优化解的能力(CS12)、易于编程实现(ES33)等。

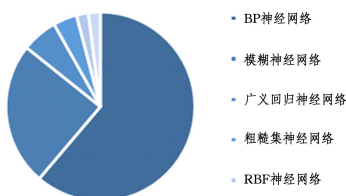


图 13 神经网络类型

## RQ3.2 采用了哪种神经网络模型？

神经网络的类型有很多,我们调研了每个评价模型使用

的神经网络类型,结果如图 13 所示。使用最多的是 BP 神经网络(30次),除了 RQ3.1 提到的神经网络的优点外, BP 神经网络还有结构简单、工作状态稳定、易于实现等优点(CS12, CS15)。

模糊神经网络同样被广泛应用(ES6, ES8, ES10, ES12, ES19, ES20, ES25, ES31, ES33, CS6, CS11, CS13)。模糊神经网络综合了模糊逻辑推理的强大结构性知识表达能力和神经网络的强大自学习能力,用模糊逻辑理论表达软件质量评价中的模糊属性,用神经网络确定质量特性的隶属度和权值,避免对软件质量特性权值确定的主观性,从而得到科学的、客观的评价(CS6)。模糊神经网络作为一种并行的计算模型,具有并行处理、自适应、联想记忆及容错和鲁棒性强等特点,有很好的非线性映射能力。由于在软件开发初期存在较少的已知信息,ES6, ES10 和 CS13 中所建立的模糊神经网络软件质量预测模型应该能够在信息不足的情况下实现软件质量属性的有效预测。另外,所获取的数据信息也可以具有不同的形式(它可以是从实际软件工程中提取的准确的历史数据,也可以是模糊的专家知识和经验 ES20),而且利用其自学习能力,可以从训练数据中自动生成易于理解并且可提取的模糊规则(ES6, ES19)。

BP 神经网络与模糊神经网络虽然被广泛应用,但是由于模糊神经网络除了拥有神经网络的特性外还汇集了模糊逻辑,因此使用两者的评价模型所评价的指标有所差异。从图 14 和图 15 中可以看出, BP 神经网络更偏重于对客观数据的处理,模糊神经网络则可以很好地学习模糊的专家知识与经验。

ES14, ES17, CS3 使用了广义回归神经网络(GRNN),其特点是人为调节的参数少,只有一个阈值,而且网络的学习全部依赖数据样本。这个特点决定了网络得以最大限度地减弱人为主观假定对预测结果的影响(CS3)。GRNN 学习速度快,网络最后收敛于样本量积聚最多的优化回归面,并且已被证明在给定足够数据的情况下在嘈杂环境中表现良好。除此之外,GRNN 的另一个优势在于它能够有效地处理稀疏数据,也就是说在数据缺乏时 GRNN 的效果较好(ES14, ES17, CS3)。

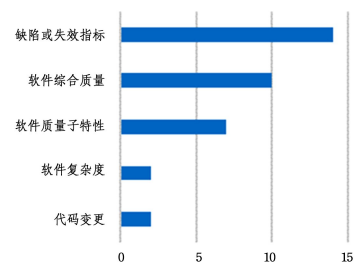


图 14 BP神经网络评价目标

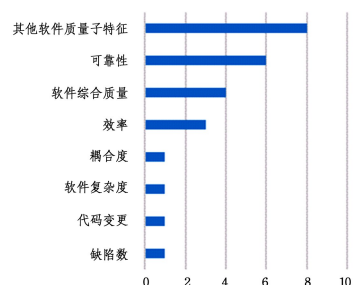


图 15 模糊神经网络评价目标

ES9 和 ES11 使用了粗糙集神经网络,粗糙集神经网络

非常适合于解决分类问题,在这些问题中,特征空间随时间的细微差别可以从粗略的近似中获得(ES9)。实践证明,粗糙神经网络能够很好地处理基于软件工程数据的决策过程中的不确定性(ES11)。

除此之外,ES12 使用了 RBF 神经网络,ES23 使用了小波神经网络。ES23 指出,小波神经网络引入了两个新的参数,即比例因子和平移因子,因此它比 RBF 神经网络具有更灵活有效的函数逼近能力、更强的模式识别与容错能力和更快的收敛速度。

### RQ3.3 哪些方法可以用于神经网络的优化?

首先我们调研了优化方法的使用对象,如图 16 所示,绝大多数优化算法都针对 BP 神经网络,这主要是因为标准 BP 网络在收敛过程中存在收敛速度比较慢、“局部最小值”问题(其梯度下降法使得权值收敛到某个值时不能保证它是误差平面的全局最小值),以及学习速率不易确定的缺点。

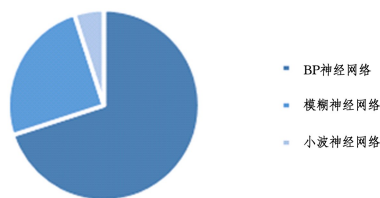


图 16 优化对象

有很多文献给出了针对性的优化方法,ES28, ES32, CS10, CS15 都使用了 LM-BP 算法来解决,LM-BP 算法将最速下降法与牛顿法相结合,既具有牛顿法的快速收敛性,又具有最速下降法的线性全局收敛特性,与标准 BP 算法相比具有较好的鲁棒性和较快的收敛速度(ES32, CS15)。

ES26 与 CS14 采用了粒子群优化算法(PSO)。CS14 指出,BP 网络采用的梯度下降法,使得权值收敛到某个值时,不能保证它是误差平面的全局最小值,而 PSO 算法避免了梯度下降法中的缺点,可以避免 BP 网络的缺陷,同时也可以缩短神经网络的训练时间。因此,用粒子群算法代替梯度下降法训练 BP 网络的参数,从而改善 BP 算法的性能。

ES6 使用了遗传算法(GA)来替代梯度下降法,因为基于梯度下降法的神经网络在学习过程开始时非常慢但最后最快,而遗传算法模拟生物进化理论在学习过程开始时非常快,但在结束时很慢,通过使用遗传算法对神经网络进行预构建,我们能够加快神经网络的学习过程。这节省了通常学习时间的 50%。

ES34 使用了教与学优化算法(Teach-Learn Based Optimization Algorithm),他们利用 TLBO 算法对神经网络的权值进行优化,可以最小化神经网络的处理时间。并且作者通过实例证明了使用教与学优化算法 TLBO 的神经网络的性能要优于使用粒子群优化算法(PSO)与遗传算法(GA)的神经网络。

除此之外,还有自调整学习率(ES23, CS1, CS8)、拟牛顿法(ES12, CS8)、启发式算法(CS7)、动量法(CS8)、混合学习算法(ES19)、元启发式算法(ES29)、混合布谷鸟搜索优化算法(ES30)、Powell-Beale 共轭梯度反向传播 CGB、弹性反向传播 RP、一步割线反向传播 OSS、比例共轭梯度反向传播 SCG(ES22)、隐马尔可夫模型(ES31)、JAYA 算法(ES35)等优化方法应用于神经网络。

### RQ3.4 除了神经网络还提供了哪些方法作为对比?

这个问题主要关心除了常用的几种神经网络模型,还有哪些方法用于软件质量评价模型建模。我们总结了使用一些对比技术进行建模的文献,总的来说软件质量评价建模技术可以分类 3 类:

第一类,传统统计方法。这类方法的目标是找到一个显式公式,如逻辑回归(ES5, ES21, CS1)、判别分析(ES1, ES5)、多元线性回归(ES15)等。

第二类,机器学习方法。这类方法尝试推导出精确可用的 if-then-else 分类规则,如案例推理(ES15)、K-近邻算法(ES21)、最小绝对偏差回归树(ES15)、最小二乘回归树(ES15)等。

第三类,神经网络方法。属于黑盒模型,除了 RQ3.2 总结的方法外还有 Ward 神经网络(ES14)、全息神经网络(ES5)等。

### RQ4 神经网络是如何训练的?

#### RQ4.1 训练数据是否需要预处理?

在对训练数据预处理之前,有些文献会对模型的输入维度进行精简,最常用的方法是主成分分析(ES1, ES2, ES3, ES4, ES5, ES7, ES10, ES15, ES23, CS3, CS5)。应用主成分分析来减少输入空间的维数并获得维度较低的输入正交域,可以增强模型的鲁棒性(ES5, ES15),主成分分析使“数据集进行了维数压缩而所含信息量却没有减少”(CS5)。其他方法还有相关性分析(ES6)、多元线性回归建模(ES3)与 FCM 聚类算法(ES21)。

在使用数据训练模型前,有些文献会对数据进行降噪(CS5)、删除冗余数据(ES24)或删除异常值(ES24)等异常数据处理。为了获得令人满意的结果,有些文献会对数据进行修正,包括重要性权重修正(CS9)与人为筛选(ES1, ES4)等。为了适应激活函数或提高模型的性能,往往对训练数据进行归一化处理(ES1, ES2, ES5, ES7, ES10, ES12, ES13, ES15, ES16, ES21, ES23, ES29, ES31, ES32, ES33, ES35, CS8, CS10, CS13, CS14, CS15)、标准化处理(ES4, CS10)、数据切割(ES10, ES18)或随机替换(ES18)等。

#### RQ4.2 训练集和测试集是如何分配的?

首先我们调查了文献训练模型所使用的数据量,其分布如图 17 所示,可以看出近一半的文献建立的神经网络模型用于训练的数据量不足 100。其次,我们调查了训练集与测试集的分配方式,我们发现训练集占总数据量的 74.6%,也就是说平均将数据按 3:1 来划分训练集与测试集。

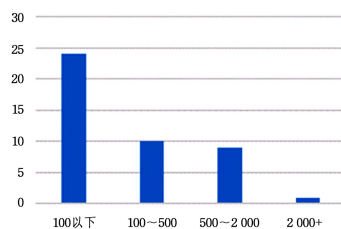


图 17 训练集数据量

#### RQ4.3 神经网络的参数是如何设置的?

神经网络因其种类不同需要设置的参数也有所不同,在这里我们主要调研了使用最多的 BP 神经网络的参数设置,主要包括网络的层数、每层的神经元数目、激活函数、代价函数与学习率的设置。

BP 神经网络常采用的层数为 3 层(18 次)与 4 层(ES22,

ES27),而模糊神经网络常采用的层数为4层(3次)与5层(4次),以对数据进行模糊化、模糊规则与去模糊化等更多处理。

输入层神经元数目等同于所采用的输入元素个数或经过主成分分析、相关性分析或者其他输入精简方法之后的输入元素个数,其分布如图18所示。输出层神经元数目等同于评价目标的数目或对评价目标等级的隶属程度,图19同样给出了它们的分布情况。

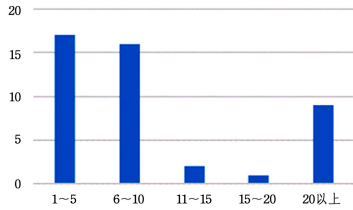


图18 输入神经元数目

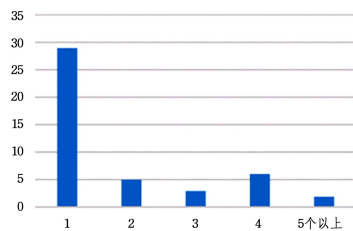


图19 输出神经元数目

对于隐含层单元数的确定,文献给出了两种方式;1)基于经验公式;2)通过预训练调整。我们调查了单隐层神经网络模型的隐含层单元数并且根据输入层、隐含层与输出层的神经元数目进行了拟合。我们的拟合结果是当输入层单元数小于10时,隐含层单元数并不会随着输入输出层的单元数变化;当输入层单元数在10与20之间时,没有发现明显的隐含层单元数与输入输出层的单元数之间的关系;而输入层单元数大于20时,隐含层单元数与输入层单元数之间表现出了很强的线性关系。因此,对于单隐层神经网络模型,当输入层单元数小于10时,可以参考的隐含层单元数为11;而当输入层单元数大于20时可以参考下面的公式计算出的隐含层单元数: $y=2.27 * x-40.39$ 。

关于神经网络的激活函数与代价函数,除了被广泛应用的sigmoid函数(22次)外,还有双曲正切tanh函数(3次)、Gaussian函数(ES21)、小波函数(ES23)等被用作神经网络的激活函数,ES16与ES27的BP神经网络的输出单元采用了恒等函数(Identity Function)。均方误差函数MSE(10次)也被广泛用于神经网络的代价函数,除此之外还有加权均方误差函数(ES22,ES27)、交叉熵函数(ES31)等。

关于学习率的设置,ES23,CS1与CS8使用了自调整学习率,而我们对其他给出具体学习率的文献进行了总结,图20给出了学习率设置的分布。

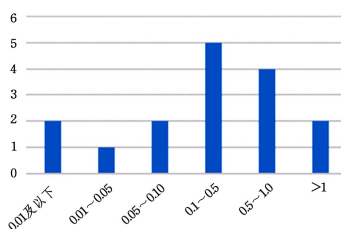


图20 学习率设置

测试集的存在是用于验证模型的性能如何,但可供选用的检验指标有很多,图21给出了常用的检验指标及其使用频率。

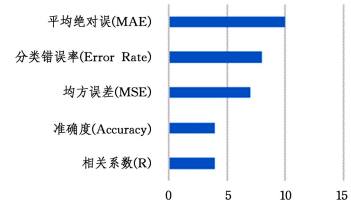


图21 验证指标

## 4 不足

本篇系统综述根据制定的研究问题尽可能详细地总结了选中的50篇文献,但此次工作存在许多有待完善之处,例如:

1)文献的质量评估是系统综述中执行环节的重要一步,但由于作者经验不足未能制定合理的质量评估准则,因此未进行文献质量评估。

2)文献的检索工作集中在了期刊与会议论文,未能包含学位论文。

3)虽然我们用粗略的关键词检索了尽可能多的数据库,但仍有可能有遗漏的相关文献没有包含在我们的综述中。可能的原因有文献检索与筛选都是由一个人完成的,无法避免遗漏与在纳入或排除文献阶段的主观偏差。

## 5 总结与展望

我们对使用神经网络方法进行软件质量评价的文献进行了一次系统综述。通过检索、筛选与研究问题相关的文献,我们最终纳入了35篇英文文献与15篇中文文献到此篇系统综述中。我们从文献的元数据以及文献中评价模型的输入元素、评价目标、建模方法以及神经网络的训练方式等方面进行了归纳总结。总而言之,应用神经网络方法的软件质量评价模型可以分为以下4种:

1)使用软件度量来预测软件缺陷、故障、失效或代码变更量(ES1, ES2, ES3, ES4, ES5, ES6, ES7, ES9, ES10, ES11, ES13, ES14, ES15, ES16, ES17, ES18, ES21, ES23, ES24, ES26, ES29, ES35, CS1, CS3, CS5, CS10);

2)使用软件度量来评价软件质量特性或其子特性(ES8, ES19, ES20, ES28, ES30, ES33, ES34, CS11, CS13, CS14);

3)使用软件质量特性或其子特性来评价软件综合质量(ES25, ES32, CS2, CS4, CS6, CS7, CS8, CS9, CS12, CS15);

4)使用软件度量来评价软件综合质量(ES12, ES22, ES27, ES31)。

软件质量评价模型最常用的神经网络是BP神经网络与模糊神经网络,虽然采用的神经网络结构简单(3~5层),训练数据的规模也都在 $10^4$ 以内,但很多学者进行了相应的算法或数据上的优化来提高神经网络的性能。

通过对文献归纳结果的分析,我们给出如下几点建议作为以后相关研究的参考:

1)进一步探究通过软件度量进行质量评价的合理性。目前文献通过软件度量的定义给出了理论性的解释(RQ1.1),但软件度量能否真正反映软件质量、哪些软件度量组合可以完全反映软件综合质量还有待验证。

2)在没有完成可以普遍评价软件质量的模型前,建立有针对性(应用领域、软件类型或者适用时期)的软件质量评价

模型。很多学者并没有对自己的评价模型给出软件类型或领域的限制,即使是其输入元素存在设计方法上的限制(RQ1.2)或训练数据只来自某一领域的软件(RQ1.3)。ES22指出软件质量评价模型因其普遍性的目标而忽视了软件类型和应用领域的限制,在软件生命周期各个阶段及决策制定过程中没有限制地使用,大大削弱了开发人员和用户对质量评价的信心。

3)探究软件缺陷与软件质量或其特性之间的关系,例如通过软件质量模型与软件缺陷分类模型对应。许多文献的软件质量评价模型通过软件度量来输出软件的缺陷、故障或失效指标,即使作者给出了用这些指标代表软件质量的原因(RQ2.1),但如果建立了这些指标与软件质量或其特性之间的关系,可以将这类评价模型与对软件综合质量或软件质量特性评价的模型联系起来,也为软件度量与软件质量之间搭建了桥梁。

4)使用较新的软件质量模型,例如 ISO-25010 软件质量模型。基于神经网络的软件质量评价模型所参考的软件质量模型大都为 20 世纪所提出的(RQ2.2),相对来说有一定的滞后性。

5)采用全面的评分方式来降低模型的主观性。大多数软件综合质量或特性评价模型的标签数据来自专家打分(RQ2.4),专家有丰富的领域经验,但对项目的细节掌握不足,可以通过软件开发与测试人员的反馈来进行一定修正。

6)增大神经网络模型训练集的数据量。许多文献训练集的数据不足 100 组(RQ4.2),如果模型没有进行合理的优化可能很容易导致过拟合,得到的结果会变差。

7)使用测试集对训练的神经网络模型进行验证,并采用合适的验证指标(RQ4.4)来检验模型的性能。部分文献将所有的数据用于训练集,再用训练集的数据进行验证,这样既无法检验模型的泛化能力,也无法检验模型是否过拟合。

**结束语** 我们对使用神经网络方法进行软件质量评价的文献进行了一次系统综述。通过检索、筛选与研究问题相关的文献,我们最终纳入了 35 篇英文文献与 15 篇中文文献到此篇系统综述中。我们从文献的元数据以及文献中评价模型的输入元素、评价目标、建模方法以及神经网络的训练方式等方面进行了归纳总结。从总结中发现,基于神经网络的软件质量评价模型建模方法已经相当成熟,缺少的是对建模数据客观性或者说可信服程度的改进;通过软件度量对软件综合质量直接进行评价的模型的建立可能较为困难,这部分研究相对很少;大部分文献的重点都放在了评价模型的建立上,很少对输入元素如何影响软件质量进行软件工程领域上的解释。希望日后的相关研究可以在评价数据的客观化、基于软件度量的综合质量评价或评价模型解释上有所突破。

## 参考文献

- [1] 崔天意,刘庆峰,张芝龙. 电子装备软件质量评估模型分析[J]. 计算机与数字工程,2010,38(1):44-47.
- [2] 马慧敏,宋雨,许正伟. 广义回归神经网络在软件质量预测中的应用[J]. 计算机工程与应用,2004(29):217-219.
- [3] MAJUMDER K. 系统综述写作指引[EB/OL]. <https://www.editage.cn/insights/1159.html>,2015
- [4] BRERETON P, KITCHENHAM B A, BUDGEN D, et al. Lessons from applying the systematic literature review process within the software engineering domain[J]. Journal of Systems and Software,2007,80(4):571-583.
- [5] KITCHENHAM B. Procedures for Undertaking Systematic Reviews[C]// Joint Technical Report, Computer Science Department, 2004.
- [6] 张家海,孙枫,郝燕玲. 神经网络用于组合导航系统软件质量的评估[J]. 哈尔滨理工大学学报,2001(3):34-37.
- [7] 赵一鸣. 基于 ISO 质量模型的软件质量评价方法[J]. 计算机工程,2002(5):78-79,82.
- [8] 杨根兴,高大启,宋国新. 基于神经网络的软件质量综合评价方法[J]. 华东理工大学学报,2004(3):292-295.
- [9] 廖锦舜,何镔. 基于模糊神经网络的软件质量评价方法[J]. 计算机技术与发展,2006(2):194-196.
- [10] 杨健,朱继梅,袁正刚. 基于 BP 网络软件质量度量模型[J]. 微计算机信息,2006(24):262-264.
- [11] 关帅,孙雅困,朱清浩. 一种潜艇作战软件质量评价的新方法[J]. 中国舰船研究,2008,3(5):54-56.
- [12] 楼文高,姜丽,孟祥辉. 计算机软件质量综合评判的人工神经网络模型[J]. 上海理工大学学报,2008(5):479-482.
- [13] 王李进,吴保国,郑德祥. 基于人工神经网络的软件质量评价[J]. 计算机应用与软件,2008,25(12):133-134,150.
- [14] 宫丽娜,马怀志. 广义动态模糊神经网络对软件质量的预测[J]. 微型电脑应用,2012,28(6):32-35.
- [15] 宫丽娜,马怀志. 粒子群算法优化的 BP 网络预测软件质量[J]. 计算机工程与应用,2014,50(23):65-68.
- [16] 窦立阳,刘建周. 基于 BP 人工神经网络的软件质量评估[J]. 计算机与数字工程,2014,42(4):644-646.
- [17] 李克文,张郁,马竞峰,等. 基于模糊三角数模糊神经网络的软件质量评价方法[J]. 计算机工程与科学,2014,36(7):1301-1306.
- [18] 郑鹏. 基于 LM-BP 神经网络的软件质量综合评价[J]. 山东理工大学学报(自然科学版),2016,30(3):74-78.
- [19] KHOSHGOFTAAR T M, LANNING D L, PANDYA A S. Comparative study of pattern recognition techniques for quality evaluation of telecommunications software[J]. IEEE Journal on Selected Areas in Communications,1994,12(2):279-291.
- [20] KHOSHGOFTAAR T M, SZABO R M. Improving neural network predictions of software quality using principal components analysis[C]// IEEE International Conference on Neural Networks,1994. IEEE World Congress on Computational Intelligence. IEEE,1994:3295-3300.
- [21] KHOSHGOFTAAR T M, SZABO R M, GUASTI P J. Exploring the behavior of neural network software quality models[J]. Software Engineering Journal,1995,10(3):89-96.
- [22] KHOSHGOFTAAR T M, ALLEN E B, HUDEPOHL J P, et al. Application of Neural Networks to Software Quality Modeling of a Very Large Telecommunications System[J]. IEEE Transactions on Neural Networks,1997,8(4):902-909.
- [23] LANUBILE F, VISAGGIO G. Evaluating predictive quality models derived from software measures: Lessons learned[J]. Journal of Systems & Software,1997,38(3):225-234.
- [24] BAISCH E, EBERT C. On a Neural-Fuzzy Technique with GA-Optimization for Software Quality Models[M]// Software Metrics,1997.
- [25] KUMAR R, RAI S, TRAHAN J L. Neural-network techniques for software-quality evaluation[C]// Reliability and Maintainability Symposium,1998. IEEE,1998:155-161.
- [26] PEDRYCZ W, PETERS J F, RAMANNA S. Software quality measurement: concepts and fuzzy neural relational model[C]// IEEE International Conference on Fuzzy Systems Proceedings,

1998. IEEE World Congress on Computational Intelligence. IEEE, 1998; 1026-1031.
- [27] PEDRYCZ W, HAN L, PETERS J F, et al. Calibration of software quality: Fuzzy neural and rough neural computing approaches[J]. *Neurocomputing*, 2001, 36(1): 149-170.
- [28] XU Z. Software Quality Prediction for High-Assurance Network Telecommunications Systems [J]. *Computer Journal*, 2001, 44(6): 557-568.
- [29] RAMANNA S, PETERS J F, AHN T C. Software Quality Knowledge Discovery: A Rough Set Approach[C]//International Computer Software & Applications Conference on Prolonging Software Life: Development & Redevelopment. IEEE Computer Society, 2002.
- [30] PIZZI N J, SUMMERS A R, PEDRYCZ W. Software quality prediction using median-adjusted class labels[C]//Neural Networks, 2002. IJCNN'02. Proceedings of the 2002 International Joint Conference. IEEE, 2002.
- [31] VALSA B. A Neural Network Model for Assessing Software Quality for Mission Critical Applications[C]//International Astronautical Congress of the International Astronautical Federation. 2003.
- [32] THWIN M M T, QUAH T S. Application of neural networks for software quality prediction using object-oriented metrics[J]. *The Journal of Systems & Software*, 2003, 76(2): 147-156.
- [33] TAGHI K, NAEEM S. Fault Prediction Modeling for Software Quality Estimation: Comparing Commonly Used Techniques [J]. *Empirical Software Engineering*, 2003, 8(3): 255-283.
- [34] WANG Q, YU B, ZHU J. Extract Rules from Software Quality Prediction Model Based on Neural Network[C]//IEEE International Conference on TOOLS with Artificial Intelligence. IEEE Computer Society, 2004; 191-195.
- [35] KANMANI S, UTHARIARAJ V R, SANKARANARAYANAN V, et al. Object-oriented software quality prediction using general regression neural networks[J]. *Acm Sigsoft Software Engineering Notes*, 2004, 29(5): 1-6.
- [36] WANG Q, YU B, YU B. Combining classifiers in software quality prediction: a neural network approach [C] // International Conference on Advances in Neural Networks. Springer-Verlag, 2005; 921-926.
- [37] YANG B, YAO L, HUANG H Z. Early Software Quality Prediction Based on a Fuzzy Neural Network Model[C]//International Conference on Natural Computation. IEEE, 2007; 760-764.
- [38] PENG W, YAO L, MIAO Q. An Approach of Software Quality Prediction Based on Relationship Analysis and Prediction Model [C]//The Proceedings of 2009, International Conference on Reliability, Maintainability and Safety. 2009; 713-717.
- [39] JIN C, JIN S W, YE J M, et al. Quality prediction model of object-oriented software system using computational intelligence [C]//International Conference on Power Electronics and Intelligent Transportation System. IEEE, 2009; 120-123.
- [40] POMOROVA O, HOVORUSHCHENKO T. Research of artificial neural network's component of software quality evaluation and prediction method[C]//IEEE International Conference on Intelligent Data Acquisition & Advanced Computing Systems. IEEE, 2011.
- [41] ZHONG C, HU Q, YANG F, et al. Software quality prediction method with hybrid applying principal components analysis and wavelet neural network and genetic algorithm[J]. *International Journal of Digital Content Technology and its Applications*, 2011, 5(3): 225-234.
- [42] RATANOTHAYANON S, SRIVISAL C, VANICHAYOBONAND S, et al. Comparative Classifiers for Software Quality Assessment[J]. *International Journal of Engineering and Technology*, 2012, 4(4): 404-408.
- [43] MIAO X, LU Y, DAI Y. Method of Military Software Quality Evaluation Based on Multi-Agent Fuzzy Neural Network[C]//International Conference on Computer Science & Service System. IEEE, 2012.
- [44] KAUR A, SINGH A, SINGH B. Design of Hybrid Neural Network Model for Quality Evaluation of Object Oriented Software Modules[J]. *International Journal of Engineering Research and Development*, 2012, 2(5): 78-82.
- [45] POMOROVA O, HOVORUSHCHENKO K. Artificial neural network for software quality evaluation based on the metric analysis[C] // East-West Design & Test Symposium. IEEE, 2013.
- [46] KUMAR G, BHATIA P K. Neuro-Fuzzy Model to Estimate & Optimize Quality and Performance of Component Based Software Engineering[J]. *Acm Sigsoft Software Engineering Notes*, 2015, 40(2): 1-6.
- [47] SURESH Y. Software quality assurance for object-oriented systems using meta-heuristic search techniques[C]//International Conference on Applied & Theoretical Computing & Communication Technology. IEEE, 2016.
- [48] SHEORAN K, TOMAR P, MISHRA R. Software Quality Prediction Model with the Aid of Advanced Neural Network with HCS[J]. *Procedia Computer Science*, 2016, 92: 418-424.
- [49] LIU Y, CHANG C, HAN P. A software development project quality analysis model based on HMM-FNN[C]//International Conference on Software Engineering and Service Science, 2017. IEEE, 2017.
- [50] WANG A, GUO L, CHEN Y, et al. Comprehensive Evaluation of Software Quality Based on LM-BP Neural Network[C]//International Conference on Dependable Systems and Their Applications. IEEE, 2017.
- [51] SAHAR S, QAMAR U, AYAZ S. Multilayer Neural Network and Fuzzy Logic Based Software Quality Prediction[J]. *International Journal of Information, Control and Computer Sciences*, 2017, 11(9): 1024-1028.
- [52] TOMAR P, MISHRA R, SHEORAN K. Prediction of quality using ANN based on Teaching-Learning Optimization in component-based software systems[J]. *Software Practice and Experience*, 2018, 48(3): 896-910.
- [53] BANSAL M, AGRAWAL C P. Jaya Algorithm and Artificial Neural Network Based Approach for Object-Oriented Software Quality Analysis[J]. *International Journal of Intelligent Engineering and Systems*, 2018, 11(4): 275-282.