



# (12)发明专利申请

(10)申请公布号 CN 110232023 A

(43)申请公布日 2019. 09. 13

(21)申请号 201910429282.X

(22)申请日 2019.05.22

(71)申请人 中国科学院长春光学精密机械与物理研究所

地址 130033 吉林省长春市长春经济技术开发区东南湖大路3888号

(72)发明人 宋元章 李洪雨 陈媛 王俊杰 王安邦

(74)专利代理机构 北京集佳知识产权代理有限公司 11227

代理人 罗满

(51)Int.Cl.

G06F 11/36(2006.01)

G06N 3/04(2006.01)

G06N 3/08(2006.01)

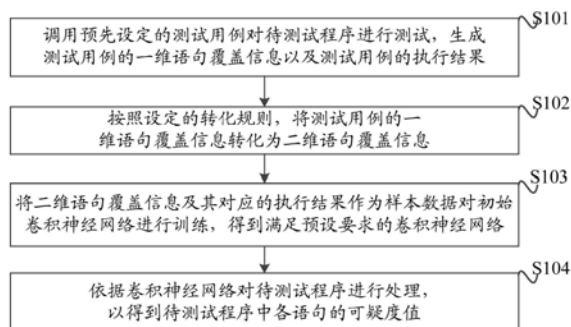
权利要求书3页 说明书13页 附图2页

## (54)发明名称

一种基于卷积神经网络的软件缺陷定位方法、装置和介质

## (57)摘要

本发明实施例公开了一种基于卷积神经网络的软件缺陷定位方法、装置和计算机可读存储介质,调用预先设定的测试用例对待测试程序进行测试,生成测试用例的一维语句覆盖信息以及测试用例的执行结果;按照设定的转化规则,将测试用例的一维语句覆盖信息转化为二维语句覆盖信息;将二维语句覆盖信息及其对应的执行结果作为样本数据对初始卷积神经网络进行训练,得到满足预设要求的卷积神经网络;通过卷积神经网络充分挖掘测试用例的一维语句覆盖信息与执行结果之间的关联关系,有效的提升了卷积神经网络对软件缺陷定位的准确性,依据卷积神经网络对待测试程序进行处理,以得到待测试程序中各语句的可疑度值,使得预测出的可疑度值更加准确。



1. 一种基于卷积神经网络的软件缺陷定位方法,其特征在于,包括:

调用预先设定的测试用例对待测试程序进行测试,生成测试用例的一维语句覆盖信息以及测试用例的执行结果;

按照设定的转化规则,将所述测试用例的一维语句覆盖信息转化为二维语句覆盖信息;

将所述二维语句覆盖信息及其对应的执行结果作为样本数据对初始卷积神经网络进行训练,得到满足预设要求的卷积神经网络;

依据所述卷积神经网络对所述待测试程序进行处理,以得到所述待测试程序中各语句的可疑度值。

2. 根据权利要求1所述的方法,其特征在于,所述按照设定的转化规则,将所述测试用例的一维语句覆盖信息转化为二维语句覆盖信息包括:

当所述测试用例的一维语句覆盖信息为稠密数据时,按照函数与测试用例的一维语句覆盖信息的对应关系,将所述测试用例的一维语句覆盖信息转化为二维语句覆盖信息;

当所述测试用例的一维语句覆盖信息为稀疏数据时,将所述测试用例的一维语句覆盖信息随机设置于二维列表中,以得到二维语句覆盖信息。

3. 根据权利要求1所述的方法,其特征在于,所述将所述二维语句覆盖信息及其对应的执行结果作为样本数据对初始卷积神经网络进行训练,得到满足预设要求的卷积神经网络包括:

将所述样本数据划分为训练样本集和测试样本集;

利用所述训练样本集对所述初始卷积神经网络进行训练,得到训练好的卷积神经网络;

将目标测试样本输入所述卷积神经网络,得到输出结果;所述目标测试样本为所述测试样本集中未经测试的任意一个测试样本;

判断所述输出结果与所述测试样本的执行结果的偏差值是否在预设范围内;

若否,则返回所述利用所述训练样本集对所述初始卷积神经网络进行训练,得到训练好的卷积神经网络的步骤;

若是,则从所述测试样本集中选取一个未经测试的测试样本作为目标测试样本,并返回所述将目标测试样本输入所述卷积神经网络,得到输出结果的步骤,直至所述测试样本集中的所有测试样本均完成测试,则将所述卷积神经网络作为满足预设要求的卷积神经网络。

4. 根据权利要求1-3任意一项所述的方法,其特征在于,所述依据所述卷积神经网络对待测试程序进行处理,以得到所述待测试程序各语句的可疑度值包括:

构建与所述待测试程序中各语句相对应的虚拟测试用例;

按照所述转化规则,将所述虚拟测试用例的一维语句覆盖信息转化为二维语句覆盖信息;

将所述二维语句覆盖信息输入所述卷积神经网络,以得到所述待测试程序各语句的可疑度值。

5. 一种基于卷积神经网络的软件缺陷定位装置,其特征在于,包括生成单元、转化单元、训练单元和评估单元;

所述生成单元,用于调用预先设定的测试用例对待测试程序进行测试,生成测试用例的一维语句覆盖信息以及测试用例的执行结果;

所述转化单元,用于按照设定的转化规则,将所述测试用例的一维语句覆盖信息转化为二维语句覆盖信息;

所述训练单元,用于将所述二维语句覆盖信息及其对应的执行结果作为样本数据对初始卷积神经网络进行训练,得到满足预设要求的卷积神经网络;

所述评估单元,用于依据所述卷积神经网络对所述待测试程序进行处理,以得到所述待测试程序中各语句的可疑度值。

6. 根据权利要求5所述的装置,其特征在于,所述转化单元包括稠密转化子单元和稀疏转化子单元;

所述稠密转化子单元,用于当所述测试用例的一维语句覆盖信息为稠密数据时,按照函数与测试用例的一维语句覆盖信息的对应关系,将所述测试用例的一维语句覆盖信息转化为二维语句覆盖信息;

所述稀疏转化子单元,用于当所述测试用例的一维语句覆盖信息为稀疏数据时,将所述测试用例的一维语句覆盖信息随机设置于二维列表中,以得到二维语句覆盖信息。

7. 根据权利要求5所述的装置,其特征在于,所述训练单元包括划分子单元、训练子单元、测试子单元、判断子单元和选取子单元;

所述划分子单元,用于将所述样本数据划分为训练样本集和测试样本集;

所述训练子单元,用于利用所述训练样本集对所述初始卷积神经网络进行训练,得到训练好的卷积神经网络;

所述测试子单元,用于将目标测试样本输入所述卷积神经网络,得到输出结果;所述目标测试样本为所述测试样本集中未经测试的任意一个测试样本;

所述判断子单元,用于判断所述输出结果与所述测试样本的执行结果的偏差值是否在预设范围内;若否,则返回所述训练子单元;若是,则触发所述选取子单元;

所述选取子单元,用于从所述测试样本集中选取一个未经测试的测试样本作为目标测试样本,并返回所述测试子单元,直至所述测试样本集中的所有测试样本均完成测试,则将所述卷积神经网络作为满足预设要求的卷积神经网络。

8. 根据权利要求5-7任意一项所述的装置,其特征在于,所述评估单元包括构建子单元、转化子单元和输出子单元;

所述构建子单元,用于构建与所述待测试程序中各语句相对应的虚拟测试用例;

所述转化子单元,用于按照所述转化规则,将所述虚拟测试用例的一维语句覆盖信息转化为二维语句覆盖信息;

所述输出子单元,用于将所述二维语句覆盖信息输入所述卷积神经网络,以得到所述待测试程序各语句的可疑度值。

9. 一种基于卷积神经网络的软件缺陷定位装置,其特征在于,包括:

存储器,用于存储计算机程序;

处理器,用于执行所述计算机程序以实现如权利要求1至4任意一项所述基于卷积神经网络的软件缺陷定位方法的步骤。

10. 一种计算机可读存储介质,其特征在于,所述计算机可读存储介质上存储有计算机

程序,所述计算机程序被处理器执行时实现如权利要求1至4任一项所述基于卷积神经网络的软件缺陷定位方法的步骤。

## 一种基于卷积神经网络的软件缺陷定位方法、装置和介质

### 技术领域

[0001] 本发明涉及软件测试技术领域,特别是涉及一种基于卷积神经网络的软件缺陷定位方法、装置和计算机可读存储介质。

### 背景技术

[0002] 软件缺陷定位旨在探测和查找引起软件失效的错误代码。传统的手工设置断点的调试方法,不仅断点位置选择困难,且时间开销巨大。因此,实现缺陷定位的自动化成为了软件学术界和工业界共同追求的目标。近些年来,研究人员从不同的角度尝试提出了一系列辅助自动化缺陷定位的方法,包括基于切片的方法、基于程序不变量的方法、模型检验方法和基于程序频谱的方法等。

[0003] 相比基于切片的方法、基于程序不变量的方法以及模型检验方法而言,基于频谱的缺陷定位方法(spectrum-based fault localization,简称SFL)由于具有不需要考虑程序本身内部结构和执行开销小的特点,成为了一类比较行之有效的重要方法。SFL方法主要通过对比分析被测程序在成功执行和失效执行的程序频谱信息,构造相应的可疑度计算公式来估测程序实体(如语句、谓词等)出错的可能性。受可疑度计算公式的局限性影响,软件缺陷定位的准确性无法达到很高。

[0004] 可见,如何提升软件缺陷定位的准确性,是本领域技术人员亟待解决的问题。

### 发明内容

[0005] 本发明实施例的目的是提供一种基于卷积神经网络的软件缺陷定位方法、装置和计算机可读存储介质,可以提升软件缺陷定位的准确性。

[0006] 为解决上述技术问题,本发明实施例提供一种基于卷积神经网络的软件缺陷定位方法,包括:

[0007] 本发明实施例还提供了一种基于卷积神经网络的软件缺陷定位方法,包括:

[0008] 调用预先设定的测试用例对待测试程序进行测试,生成测试用例的一维语句覆盖信息以及测试用例的执行结果;

[0009] 按照设定的转化规则,将所述测试用例的一维语句覆盖信息转化为二维语句覆盖信息;

[0010] 将所述二维语句覆盖信息及其对应的执行结果作为样本数据对初始卷积神经网络进行训练,得到满足预设要求的卷积神经网络;

[0011] 依据所述卷积神经网络对所述待测试程序进行处理,以得到所述待测试程序中各语句的可疑度值。

[0012] 可选的,所述按照设定的转化规则,将所述测试用例的一维语句覆盖信息转化为二维语句覆盖信息包括:

[0013] 当所述测试用例的一维语句覆盖信息为稠密数据时,按照函数与测试用例的一维语句覆盖信息的对应关系,将所述测试用例的一维语句覆盖信息转化为二维语句覆盖信

息；

[0014] 当所述测试用例的一维语句覆盖信息为稀疏数据时，将所述测试用例的一维语句覆盖信息随机设置于二维列表中，以得到二维语句覆盖信息。

[0015] 可选的，所述将所述二维语句覆盖信息及其对应的执行结果作为样本数据对初始卷积神经网络进行训练，得到满足预设要求的卷积神经网络包括：

[0016] 将所述样本数据划分为训练样本集和测试样本集；

[0017] 利用所述训练样本集对所述初始卷积神经网络进行训练，得到训练好的卷积神经网络；

[0018] 将目标测试样本输入所述卷积神经网络，得到输出结果；所述目标测试样本为所述测试样本集中未经测试的任意一个测试样本；

[0019] 判断所述输出结果与所述测试样本的执行结果的偏差值是否在预设范围内；

[0020] 若否，则返回所述利用所述训练样本集对所述初始卷积神经网络进行训练，得到训练好的卷积神经网络的步骤；

[0021] 若是，则从所述测试样本集中选取一个未经测试的测试样本作为目标测试样本，并返回所述将目标测试样本输入所述卷积神经网络，得到输出结果的步骤，直至所述测试样本集中的所有测试样本均完成测试，则将所述卷积神经网络作为满足预设要求的卷积神经网络。

[0022] 可选的，所述依据所述卷积神经网络对待测试程序进行处理，以得到所述待测试程序各语句的可疑度值包括：

[0023] 构建与所述待测试程序中各语句相对应的虚拟测试用例；

[0024] 按照所述转化规则，将所述虚拟测试用例的一维语句覆盖信息转化为二维语句覆盖信息；

[0025] 将所述二维语句覆盖信息输入所述卷积神经网络，以得到所述待测试程序各语句的可疑度值。

[0026] 本发明实施例还提供了一种基于卷积神经网络的软件缺陷定位装置，包括生成单元、转化单元、训练单元和评估单元；

[0027] 所述生成单元，用于调用预先设定的测试用例对待测试程序进行测试，生成测试用例的一维语句覆盖信息以及测试用例的执行结果；

[0028] 所述转化单元，用于按照设定的转化规则，将所述测试用例的一维语句覆盖信息转化为二维语句覆盖信息；

[0029] 所述训练单元，用于将所述二维语句覆盖信息及其对应的执行结果作为样本数据对初始卷积神经网络进行训练，得到满足预设要求的卷积神经网络；

[0030] 所述评估单元，用于依据所述卷积神经网络对所述待测试程序进行处理，以得到所述待测试程序中各语句的可疑度值。

[0031] 可选的，所述转化单元包括稠密转化子单元和稀疏转化子单元；

[0032] 所述稠密转化子单元，用于当所述测试用例的一维语句覆盖信息为稠密数据时，按照函数与测试用例的一维语句覆盖信息的对应关系，将所述测试用例的一维语句覆盖信息转化为二维语句覆盖信息；

[0033] 所述稀疏转化子单元，用于当所述测试用例的一维语句覆盖信息为稀疏数据时，

将所述测试用例的一维语句覆盖信息随机设置于二维列表中,以得到二维语句覆盖信息。

[0034] 可选的,所述训练单元包括划分子单元、训练子单元、测试子单元、判断子单元和选取子单元;

[0035] 所述划分子单元,用于将所述样本数据划分为训练样本集和测试样本集;

[0036] 所述训练子单元,用于利用所述训练样本集对所述初始卷积神经网络进行训练,得到训练好的卷积神经网络;

[0037] 所述测试子单元,用于将目标测试样本输入所述卷积神经网络,得到输出结果;所述目标测试样本为所述测试样本集中未经测试的任意一个测试样本;

[0038] 所述判断子单元,用于判断所述输出结果与所述测试样本的执行结果的偏差值是否在预设范围内;若否,则返回所述训练子单元;若是,则触发所述选取子单元;

[0039] 所述选取子单元,用于从所述测试样本集中选取一个未经测试的测试样本作为目标测试样本,并返回所述测试子单元,直至所述测试样本集中的所有测试样本均完成测试,则将所述卷积神经网络作为满足预设要求的卷积神经网络。

[0040] 可选的,所述评估单元包括构建子单元、转化子单元和输出子单元;

[0041] 所述构建子单元,用于构建与所述待测试程序中各语句相对应的虚拟测试用例;

[0042] 所述转化子单元,用于按照所述转化规则,将所述虚拟测试用例的一维语句覆盖信息转化为二维语句覆盖信息;

[0043] 所述输出子单元,用于将所述二维语句覆盖信息输入所述卷积神经网络,以得到所述待测试程序各语句的可疑度值。

[0044] 本发明实施例还提供了一种基于卷积神经网络的软件缺陷定位装置,包括:

[0045] 存储器,用于存储计算机程序;

[0046] 处理器,用于执行所述计算机程序以实现如上述基于卷积神经网络的软件缺陷定位方法的步骤。

[0047] 本发明实施例还提供了一种计算机可读存储介质,所述计算机可读存储介质上存储有计算机程序,所述计算机程序被处理器执行时实现如上述基于卷积神经网络的软件缺陷定位方法的步骤。

[0048] 由上述技术方案可以看出,调用预先设定的测试用例对待测试程序进行测试,生成测试用例的一维语句覆盖信息以及测试用例的执行结果;为了与卷积神经网络的数据格式相适应,需要按照设定的转化规则,将测试用例的一维语句覆盖信息转化为二维语句覆盖信息;将二维语句覆盖信息及其对应的执行结果作为样本数据对初始卷积神经网络进行训练,得到满足预设要求的卷积神经网络;通过卷积神经网络充分挖掘测试用例的一维语句覆盖信息与执行结果之间的关联关系,有效的提升了卷积神经网络对软件缺陷定位的准确性,当需要对待测试程序分析其软件缺陷时,则可以依据卷积神经网络对待测试程序进行处理,以得到待测试程序中各语句的可疑度值。基于卷积神经网络有效的挖掘测试用例的一维语句覆盖信息与执行结果之间的关联关系,使得预测出的可疑度值更加准确。

## 附图说明

[0049] 为了更清楚地说明本发明实施例,下面将对实施例中所需要使用的附图做简单的介绍,显而易见地,下面描述中的附图仅仅是本发明的一些实施例,对于本领域普通技术人

员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0050] 图1为本发明实施例提供的一种基于卷积神经网络的软件缺陷定位方法的流程图;

[0051] 图2为本发明实施例提供的一种基于卷积神经网络的软件缺陷定位装置的结构示意图;

[0052] 图3为本发明实施例提供的一种基于卷积神经网络的软件缺陷定位装置的结构示意图。

## 具体实施方式

[0053] 下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本发明一部分实施例,而不是全部实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动前提下,所获得的所有其他实施例,都属于本发明保护范围。

[0054] 为了使本技术领域的人员更好地理解本发明方案,下面结合附图和具体实施方式对本发明作进一步的详细说明。

[0055] 软件缺陷定位主要目的在于探测和查找引起软件失效的错误代码。传统方法中,往往引用其他相关领域计算公式,并进行大量实验研究来设计缺陷定位的可疑度计算公式;或者是从被测程序本身入手,通过挖掘程序自身蕴含的一些规律信息,例如事件包含的信息量、执行轨迹模式等来构造出新的频谱方法。基于该前提,研究人员应用测试用例执行产生的语句覆盖信息和测试用例的执行结果来构造可疑度计算公式Tarantula,从而辅助软件缺陷定位。Tarantula公式中定义了4类因子用以描述测试用例执行产生的语句覆盖信息和执行结果之间的关系: $a_{11}$ 表示某条语句在失效测试中覆盖到的次数; $a_{10}$ 表示某条语句在成功测试中覆盖到的次数; $a_{00}$ 表示某条语句在成功测试中未被覆盖到的次数; $a_{01}$ 表示某条语句在失效测试中未被覆盖到的次数。

[0056] 传统缺陷定位方法效果差异的部分原因可以归结为:如何利用 $a_{11}$ 、 $a_{10}$ 、 $a_{00}$ 、 $a_{01}$ 这4个权重因子蕴含的信息来计算语句的可疑度。根据测试用例执行产生的语句覆盖信息与执行结果之间潜在关联的假设,对这4个因子的权重考虑得越准确,那么对应方法的缺陷定位效果也就越好。

[0057] 测试用例执行产生的语句覆盖信息和执行结果之间存在的潜在关联是确定可疑度的基础。显然,充分地挖掘和利用这种潜在关联蕴含的缺陷揭示信息,有助于提升可疑度计算公式缺陷定位的效用。因此,本发明实施例提供的一种基于卷积神经网络的软件缺陷定位方法、装置和计算机可读存储介质,利用卷积神经网络来充分挖掘测试用例执行产生的语句覆盖信息和执行结果之间的潜在关系,将测试用例执行产生的语句覆盖信息和测试用例的执行结果分别作为样本的特征数据、标记数据对卷积神经网络进行训练,有效的提升了卷积神经网络测试结果的准确性。

[0058] 接下来,详细介绍本发明实施例所提供的一种基于卷积神经网络的软件缺陷定位方法。图1为本发明实施例提供的一种基于卷积神经网络的软件缺陷定位方法的流程图,该方法包括:

[0059] S101:调用预先设定的测试用例对待测试程序进行测试,生成测试用例的一维语



句覆盖信息以及测试用例的执行结果。

[0060] 利用测试用例在待测试程序中运行,当测试用例遍历到待测试程序中的某个语句时,则表示测试用例覆盖了该语句,在本发明实施例中,可以用测试用例的一维语句覆盖信息表示测试用例对待测试程序的覆盖情况。

[0061] 为了便于描述,可以用 $P = \{s_1, s_2, \dots, s_z\}$ 表示待测试程序。其中, $s_i (1 \leq i \leq z)$ 表示第 $i$ 个语句, $z$ 表示构成该待测试程序的语句的个数。

[0062] 待测试程序一般由多个函数构成,每个函数包含有多个语句。以函数形式表示测试用例 $P = \{f_1, f_2, \dots, f_m\}$ 。其中, $f_i (1 \leq i \leq m)$ 表示第 $i$ 个函数, $m$ 表示构成程序的函数的个数。

[0063]  $f_i = \{E_{i1}, E_{i2}, \dots, E_{in}\}$ 表示函数 $f_i$ 中语句的集合。其中, $E_{ij} (1 \leq i \leq m, 1 \leq j \leq n)$ 表示函数 $f_i$ 的第 $j$ 条语句, $n$ 表示函数 $f_i$ 的语句的总数。

[0064]  $T = \{t_1, t_2, \dots, t_l\}$ 表示测试用例的集合。其中, $t_i (1 \leq i \leq l)$ 表示第 $i$ 个测试用例。

[0065]  $Ps(t_i) = \{e_1, e_2, \dots, e_j, \dots, e_z\}$ 表示测试用例 $t_i (1 \leq i \leq l)$ 执行产生的测试用例的一维语句覆盖信息。其中, $e_j (1 \leq j \leq z)$ 表示 $t_i$ 是否覆盖语句 $s_j$ ,若 $t_i$ 覆盖语句 $s_j$ ,则 $e_j$ 值为1,否则 $e_j$ 值为0。

[0066]  $Rs(t_i)$ 表示测试用例 $t_i (1 \leq i \leq l)$ 的执行结果,若 $t_i$ 执行成功,则 $Rs(t_i)$ 为0,若 $t_i$ 执行失败,则 $Rs(t_i)$ 为1。

[0067] 以一个待测试程序为例,其对应的语句覆盖信息和测试用例执行结果如表1所示:

	待测试程序的语句	测试用例		
		$t_1$	$t_2$	$t_3$
[0068]	$s_1$	1	0	0
	$s_2$	1	1	0
	$s_3$	0	1	0
	$s_4$	1	1	1
	$s_5$	0	0	1
	$s_6$	0	1	0
	测试用例执行结果	1	0	1

[0069] 表1

[0070] 如表2所示表示函数与语句的对应关系:

[0071]	语句	函数	按照函数划分后的一维语句覆盖信息
	$s_1$	$f_1$	$E_{11}$
	$s_2$		$E_{12}$
	$s_3$		$E_{13}$
	$s_4$	$f_2$	$E_{21}$
	$s_5$		$E_{22}$
	$s_6$	$f_3$	$E_{31}$

[0072] 表2

[0073] S102:按照设定的转化规则,将测试用例的一维语句覆盖信息转化为二维语句覆盖信息。

[0074] 在本发明实施例中,利用卷积神经网络来挖掘测试用例执行产生的语句覆盖信息和执行结果之间的潜在关系。考虑到卷积神经网络用于对高维数据进行处理,因此,需要将得到的测试用例的一维语句覆盖信息转化为二维语句覆盖信息。

[0075] 在本发明实施例中,可以采用两种不同的转化方式,将测试用例的一维语句覆盖信息转化为二维语句覆盖信息。

[0076] 具体的,可以当测试用例的一维语句覆盖信息为稠密数据时,按照函数与测试用例的一维语句覆盖信息的对应关系,将测试用例的一维语句覆盖信息转化为二维语句覆盖信息;当测试用例的一维语句覆盖信息为稀疏数据时,将测试用例的一维语句覆盖信息随机设置于二维列表中,以得到二维语句覆盖信息。

[0077] 需要说明的是,在实际应用中,对于稀疏数据,也可以按照函数与测试用例的一维语句覆盖信息的对应关系,将测试用例的一维语句覆盖信息转化为二维语句覆盖信息;对于稠密数据,也可以将测试用例的一维语句覆盖信息随机设置于二维列表中,以得到二维语句覆盖信息。在本发明实施例中,对于具体采用哪种转化方式不做限定。

[0078] 假设转换后的二维数据的列数为 $D_h$ 、行数为 $D_l$ 。当测试用例的一维语句覆盖信息较为稠密时,按照表2中函数与语句的对应关系,可以令 $D_h$ 等于函数的个数 $m$ , $D_l$ 等于“具有最多语句条数的函数”的语句条数,每列的数据表示某函数的语句的覆盖信息。将表1中的测试用例的一维语句覆盖信息转换为二维语句覆盖信息的初始格式如下表3a所示:

[0079]

$E_{11}$	$E_{21}$	$E_{31}$
$E_{12}$	$E_{22}$	填充
$E_{13}$	填充	填充

[0080] 表3a

[0081] 当 $D_h \times D_l \neq z$ 时,需要对 $D_h \times D_l - z$ 个空格子进行数据填充。其中, $z$ 表示样本数据中语句的个数。

[0082] 填充的数据可以符合随机正态分布、均匀分布、直接填充有效范围的最小值。不失一般性,可以将待填充的数据设置为0,相应的,对表3a进行填充后的形式如下表所示:

[0083] 以表3a所示的格式为例,测试用例 $t_1$ 对应的二维语句覆盖信息如下表3b 所示,其对应的执行结果为1:

[0084]

1	1	0
1	0	0
0	0	0

[0085] 表3b

[0086] 测试用例 $t_2$ 对应的二维语句覆盖信息如下表3c所示,其对应的执行结果为0:

[0087]

0	1	1
1	0	0
1	0	0

[0088] 表3c

[0089] 测试用例 $t_3$ 对应的二维语句覆盖信息如下表3d所示,其对应的执行结果为1:

[0090]

0	1	0
0	1	0
0	0	0

[0091] 表3d

[0092] 当测试用例的一维语句覆盖信息较为稀疏时,可以将一维语句覆盖数据随机落在 $Dh \times Dl$ 个格子的 $z$ 个格子里, $z \leq Dh \times Dl \leq \left(\left\lfloor \sqrt{z} \right\rfloor + 1\right)^2$ ,其中 $\left\lfloor \sqrt{z} \right\rfloor \leq Dh \leq Dl \leq \left\lceil \sqrt{z} \right\rceil$ 。

[0093] 将表1中的测试用例的一维语句覆盖信息转换为二维语句覆盖信息的一种初始格式如下表4a所示:

[0094]

$E_{22}$	$E_{21}$	$E_{13}$
$E_{12}$	$E_{11}$	$E_{31}$

[0095] 表4a

[0096] 需要说明的是,因为是随机设置,所以表4a的形式仅是一种可能的表格形式。

[0097] 以表4a所示的格式为例,测试用例 $t_1$ 对应的二维语句覆盖信息如下表4b 所示,其对应的执行结果为1:

[0098]

0	1	0
1	1	0

[0099] 表4b

[0100] 测试用例 $t_2$ 对应的二维语句覆盖信息如下表4c所示,其对应的执行结果为0:

[0101]

0	1	1
---	---	---

1	0	1
---	---	---

[0102] 表4c

[0103] 测试用例 $t_3$ 对应的二维语句覆盖信息如下表4d所示,其对应的执行结果为1:

[0104]

1	1	0
0	0	0

[0105] 表4d

[0106] S103:将二维语句覆盖信息及其对应的执行结果作为样本数据对初始卷积神经网络进行训练,得到满足预设要求的卷积神经网络。

[0107] 二维语句覆盖信息即为卷积神经网络可识别的测试用例的语句覆盖信息形式。

[0108] 在本发明实施例中,将样本数据划分为训练样本集和测试样本集;利用训练样本集对初始卷积神经网络进行训练,得到训练好的卷积神经网络。

[0109] 其中,卷积神经网络主要由输入层、卷积层、池化(Pooling)层、全连接层和输出层这几类层构成。通过将这些层叠加起来,就可以构建一个完整的卷积神经网络。

[0110] 在本发明实施例中,构建的卷积神经网络的拓扑结构如下所示:

[0111] (1) 输入层Input

[0112] 输入层的层数为1层,输入为 $D_h \times D_l$ 的二维数据。

[0113] (2) 卷积层

[0114] 构建卷积层时包括确定卷积层的层数;确定每个卷积层的卷积核个数、卷积核尺寸、卷积核滑动的步长;确定是否加入偏置(Nobias);确定是否采用宽卷积运算(卷积运算前对边缘进行补0);在激活函数激活前,确定是否加入 L2正则化对卷积核参数本身进行惩罚(L2正则参数)。

[0115] (3) 池化层

[0116] 构建池化层时包括确定池化层的层数;确定每个池化层的池化窗尺寸、池化窗滑动的步长、池化操作的形式(均值池化、最大值池化等)。

[0117] (4) 全连接层

[0118] 构建全连接层时包括确定全连接层的层数;确定每个连接层的Dropout 参数。

[0119] (5) 输出层

[0120] 输出层的层数为一层,节点数为1。输出层使用逻辑函数或归一化指数函数(softmax function)输出分类标签。

[0121] 在本发明实施例中,各层的激活函数可以采用修正线性单元(Rectified Linear Unit,ReLU)函数。学习算法可以采用反向传播算法(Back Propagation Algorithm,BPA)。

[0122] 以测试样本集中未经测试的任意一个测试样本即目标测试样本为例,可以将目标测试样本输入卷积神经网络,得到输出结果。判断输出结果与测试样本的执行结果的偏差值是否在预设范围内;若否,则返回利用训练样本集对初始卷积神经网络进行训练,得到训练好的卷积神经网络的步骤;若是,则从测试样本集中选取一个未经测试的测试样本作为目标测试样本,并返回将目标测试样本输入卷积神经网络,得到输出结果的步骤,直至测试样本集中的所有测试样本均完成测试,则将卷积神经网络作为满足预设要求的卷积神经网络。

[0123] 例如,将测试样本对应的二维语句覆盖信息输入到卷积神经网络后得到的输出结果为 $Y'$ ,该测试样本对应的测试用例执行结果为 $Y$ ,当满足  $|Y'-Y| \leq \tau$ ,则认为输出结果与测试样本的执行结果的偏差值在预设范围内。否则,认为输出结果与测试样本的执行结果的偏差值不在预设范围内。

[0124] S104:依据卷积神经网络对待测试程序进行处理,以得到待测试程序中各语句的可疑度值。

[0125] 当训练好卷积神经网络之后,便可以依据该卷积神经网络对待测试程序进行缺陷定位。考虑到卷积神经网络训练时输入的是测试用例的二维语句覆盖信息的形式,相应的,在对待测试程序进行缺陷定位时,需要构建与待测试程序中各语句相对应的虚拟测试用例;按照转化规则,将虚拟测试用例的一维语句覆盖信息转化为二维语句覆盖信息;将二维语句覆盖信息输入卷积神经网络,以得到待测试程序各语句的可疑度值。

[0126] 在构建虚拟测试用例时,可以针对于每个语句构建其对应的一个虚拟测试用例,将一个虚拟测试用例在待测试程序中运行时,只会调用到待测试程序中相应的一个语句。

[0127] 以待测试程序中包含6个语句为例,依次为 $s_1$ 至 $s_6$ 。

[0128] 语句 $s_1$ 对应的虚拟测试用例 $ts_1$ 的一维语句覆盖信息为  $Ps'(ts_1) = \{1, 0, 0, 0, 0, 0\}$ 。

[0129] 语句 $s_2$ 对应的虚拟测试用例 $ts_2$ 的一维语句覆盖信息为  $Ps'(ts_2) = \{0, 1, 0, 0, 0, 0\}$ 。

[0130] 语句 $s_3$ 对应的虚拟测试用例 $ts_3$ 的一维语句覆盖信息为  $Ps'(ts_3) = \{0, 0, 1, 0, 0, 0\}$ 。

[0131] 语句 $s_4$ 对应的虚拟测试用例 $ts_4$ 的一维语句覆盖信息为  $Ps'(ts_4) = \{0, 0, 0, 1, 0, 0\}$ 。

[0132] 语句 $s_5$ 对应的虚拟测试用例 $ts_5$ 的一维语句覆盖信息为  $Ps'(ts_5) = \{0, 0, 0, 0, 1, 0\}$ 。

[0133] 语句 $s_6$ 对应的虚拟测试用例 $ts_6$ 的一维语句覆盖信息为  $Ps'(ts_6) = \{0, 0, 0, 0, 0, 1\}$ 。

[0134] (1)采用稠密处理方式,即按照函数与测试用例的一维语句覆盖信息的对应关系,将测试用例的一维语句覆盖信息转化为二维语句覆盖信息的格式如下所示:

[0135] 语句 $s_1$ 的虚拟测试用例 $ts_1$ 对应的卷积神经网络的输入数据:

[0136]

1	0	0
0	0	0
0	0	0

[0137] 语句 $s_2$ 的虚拟测试用例 $ts_2$ 对应的卷积神经网络的输入数据:

[0138]

0	0	0
1	0	0
0	0	0

[0139] 语句 $s_3$ 的虚拟测试用例 $ts_3$ 对应的卷积神经网络的输入数据:

[0140]

0	0	0
0	0	0
1	0	0

[0141] 语句 $s_4$ 的虚拟测试用例 $t_{s_4}$ 对应的卷积神经网络的输入数据:

[0142]

0	1	0
0	0	0
0	0	0

[0143] 语句 $s_5$ 的虚拟测试用例 $t_{s_5}$ 对应的卷积神经网络的输入数据:

[0144]

0	0	0
0	1	0
0	0	0

[0145] 语句 $s_6$ 的虚拟测试用例 $t_{s_6}$ 对应的卷积神经网络的输入数据:

[0146]

0	0	1
0	0	0
0	0	0

[0147] (2) 采用稀疏处理方式,即将测试用例的一维语句覆盖信息随机设置于二维列表中,以得到二维语句覆盖信息的格式如下所示:

[0148] 语句 $s_1$ 的虚拟测试用例 $t_{s_1}$ 对应的卷积神经网络的输入数据:

[0149]

0	0	0
0	1	0

[0150] 语句 $s_2$ 的虚拟测试用例 $t_{s_2}$ 对应的卷积神经网络的输入数据:

[0151]

0	0	0
1	0	0

[0152] 语句 $s_3$ 的虚拟测试用例 $t_{s_3}$ 对应的卷积神经网络的输入数据:

[0153]

0	0	1
0	0	0

[0154] 语句 $s_4$ 的虚拟测试用例 $t_{s_4}$ 对应的卷积神经网络的输入数据:

[0155]

0	1	0
0	0	0

[0156] 语句 $s_5$ 的虚拟测试用例 $t_{s_5}$ 对应的卷积神经网络的输入数据:

[0157]

1	0	0
---	---	---

0	0	0
---	---	---

[0158] 语句s<sub>6</sub>的虚拟测试用例t<sub>s<sub>6</sub></sub>对应的卷积神经网络的输入数据:

[0159]

0	0	0
0	0	1

[0160] 当确定出待测试程序中各语句的可疑度值时,便可以根据各语句的可疑度值确定出哪些语句为问题语句,从而实现缺陷的定位。

[0161] 由上述技术方案可以看出,调用预先设定的测试用例对待测试程序进行测试,生成测试用例的一维语句覆盖信息以及测试用例的执行结果;按照设定的转化规则,将测试用例的一维语句覆盖信息转化为二维语句覆盖信息;将二维语句覆盖信息及其对应的执行结果作为样本数据对初始卷积神经网络进行训练,得到满足预设要求的卷积神经网络;通过卷积神经网络充分挖掘测试用例的一维语句覆盖信息与执行结果之间的关联关系,有效的提升了卷积神经网络对软件缺陷定位的准确性,当需要对待测试程序分析其软件缺陷时,则可以依据卷积神经网络对待测试程序进行处理,以得到待测试程序中各语句的可疑度值。基于卷积神经网络有效的挖掘测试用例的一维语句覆盖信息与执行结果之间的关联关系,使得预测出的可疑度值更加准确。

[0162] 卷积神经网络输出的各语句的可疑度值的取值范围为0至1。在本发明实施例中,在得到待测试程序中各语句的可疑度值之后还可以将待测试程序的各语句按照可疑度值降序的顺序排列,生成软件缺陷定位报告表。以S104中介绍的待测试程序中包含6个语句为例,其对应的软件缺陷定位报告表如下所示:

[0163]

可疑度值	语句s <sub>i</sub>
0.93	S <sub>1</sub>
0.91	S <sub>2</sub>
0.91	S <sub>3</sub>
0.76	S <sub>4</sub>
0.53	S <sub>5</sub>
0.18	S <sub>6</sub>

[0164] 软件缺陷定位报告表

[0165] 通过对待测试程序中各语句的可疑度值进行降序排列,得到软件缺陷定位报告表,可以便于相关人员直观的了解待测试程序中各语句的缺陷情况,从而实现对缺陷语句的快速定位处理。

[0166] 图2为本发明实施例提供的一种基于卷积神经网络的软件缺陷定位装置的结构示意图,包括生成单元21、转化单元22、训练单元23和评估单元24;

[0167] 生成单元21,用于调用预先设定的测试用例对待测试程序进行测试,生成测试用例的一维语句覆盖信息以及测试用例的执行结果;

[0168] 转化单元22,用于按照设定的转化规则,将测试用例的一维语句覆盖信息转化为二维语句覆盖信息;

[0169] 训练单元23,用于将二维语句覆盖信息及其对应的执行结果作为样本数据对初始

卷积神经网络进行训练,得到满足预设要求的卷积神经网络;

[0170] 评估单元24,用于依据卷积神经网络对待测试程序进行处理,以得到待测试程序中各语句的可疑度值。

[0171] 可选的,转化单元包括稠密转化子单元和稀疏转化子单元;

[0172] 稠密转化子单元,用于当测试用例的一维语句覆盖信息为稠密数据时,按照函数与测试用例的一维语句覆盖信息的对应关系,将测试用例的一维语句覆盖信息转化为二维语句覆盖信息;

[0173] 稀疏转化子单元,用于当测试用例的一维语句覆盖信息为稀疏数据时,将测试用例的一维语句覆盖信息随机设置于二维列表中,以得到二维语句覆盖信息。

[0174] 可选的,训练单元包括划分子单元、训练子单元、测试子单元、判断子单元和选取子单元;

[0175] 划分子单元,用于将样本数据划分为训练样本集和测试样本集;

[0176] 训练子单元,用于利用训练样本集对初始卷积神经网络进行训练,得到训练好的卷积神经网络;

[0177] 测试子单元,用于将目标测试样本输入卷积神经网络,得到输出结果;目标测试样本为测试样本集中未经测试的任意一个测试样本;

[0178] 判断子单元,用于判断输出结果与测试样本的执行结果的偏差值是否在预设范围内;若否,则返回训练子单元;若是,则触发选取子单元;

[0179] 选取子单元,用于从测试样本集中选取一个未经测试的测试样本作为目标测试样本,并返回测试子单元,直至测试样本集中的所有测试样本均完成测试,则将卷积神经网络作为满足预设要求的卷积神经网络。

[0180] 可选的,评估单元包括构建子单元、转化子单元和输出子单元;

[0181] 构建子单元,用于构建与待测试程序中各语句相对应的虚拟测试用例;

[0182] 转化子单元,用于按照转化规则,将虚拟测试用例的一维语句覆盖信息转化为二维语句覆盖信息;

[0183] 输出子单元,用于将二维语句覆盖信息输入卷积神经网络,以得到待测试程序各语句的可疑度值。

[0184] 图2所对应实施例中特征的说明可以参见图1所对应实施例的相关说明,这里不再一一赘述。

[0185] 由上述技术方案可以看出,调用预先设定的测试用例对待测试程序进行测试,生成测试用例的一维语句覆盖信息以及测试用例的执行结果;为了与卷积神经网络的数据格式相适应,需要按照设定的转化规则,将测试用例的一维语句覆盖信息转化为二维语句覆盖信息;将二维语句覆盖信息及其对应的执行结果作为样本数据对初始卷积神经网络进行训练,得到满足预设要求的卷积神经网络;通过卷积神经网络充分挖掘测试用例的一维语句覆盖信息与执行结果之间的关联关系,有效的提升了卷积神经网络对软件缺陷定位的准确性,当需要对待测试程序分析其软件缺陷时,则可以依据卷积神经网络对待测试程序进行处理,以得到待测试程序中各语句的可疑度值。基于卷积神经网络有效的挖掘测试用例的一维语句覆盖信息与执行结果之间的关联关系,使得预测出的可疑度值更加准确。

[0186] 图3为本发明实施例提供一种基于卷积神经网络的软件缺陷定位装置 30的结



构示意图,包括:

[0187] 存储器31,用于存储计算机程序;

[0188] 处理器32,用于执行计算机程序以实现如上述基于卷积神经网络的软件缺陷定位方法的步骤。

[0189] 本发明实施例还提供了一种计算机可读存储介质,计算机可读存储介质上存储有计算机程序,计算机程序被处理器执行时实现如上述基于卷积神经网络的软件缺陷定位方法的步骤。

[0190] 以上对本发明实施例所提供的一种基于卷积神经网络的软件缺陷定位方法、装置和计算机可读存储介质进行了详细介绍。说明书中各个实施例采用递进的方式描述,每个实施例重点说明的都是与其他实施例的不同之处,各个实施例之间相同相似部分互相参见即可。对于实施例公开的装置而言,由于其与实施例公开的方法相对应,所以描述的比较简单,相关之处参见方法部分说明即可。应当指出,对于本技术领域的普通技术人员来说,在不脱离本发明原理的前提下,还可以对本发明进行若干改进和修饰,这些改进和修饰也落入本发明权利要求的保护范围内。

[0191] 专业人员还可以进一步意识到,结合本文中所公开的实施例描述的各示例的单元及算法步骤,能够以电子硬件、计算机软件或者二者的结合来实现,为了清楚地说明硬件和软件的可互换性,在上述说明中已经按照功能一般性地描述了各示例的组成及步骤。这些功能究竟以硬件还是软件方式来执行,取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能,但是这种实现不应认为超出本发明的范围。

[0192] 结合本文中所公开的实施例描述的方法或算法的步骤可以直接用硬件、处理器执行的软件模块,或者二者的结合来实施。软件模块可以置于随机存储器(RAM)、内存、只读存储器(ROM)、电可编程ROM、电可擦除可编程ROM、寄存器、硬盘、可移动磁盘、CD-ROM、或技术领域内所公知的任意其它形式的存储介质中。

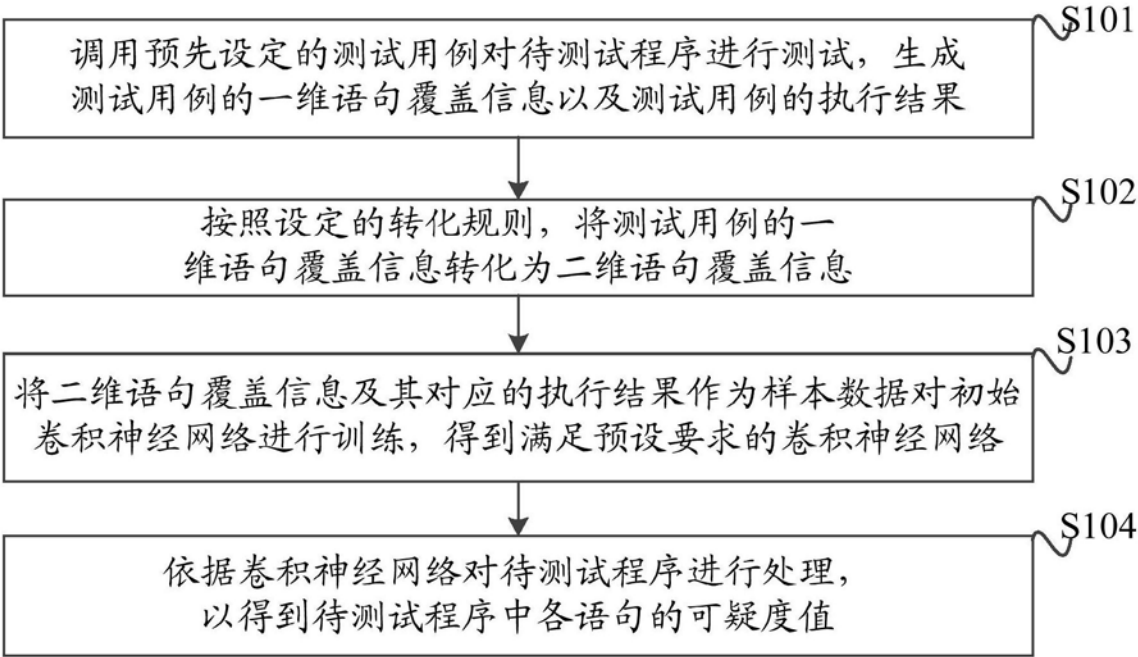


图1

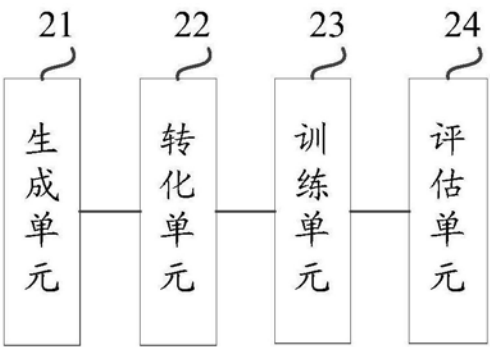


图2

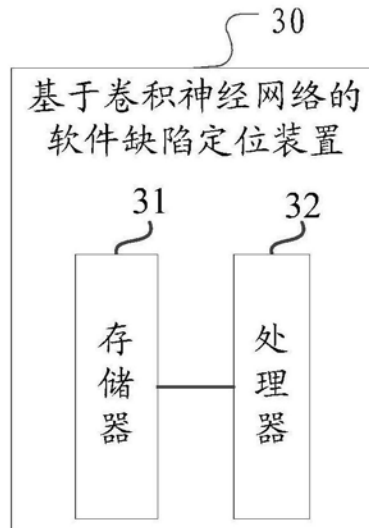


图3