TEST DATA GENERATION METHOD AND APPARATUS, MEDIUM, AND COMPUTER SYSTEM

说明书

一种测试数据生成方法、装置、介质及计算机系统

技术领域

本说明书涉及计算机技术领域，尤其涉及一种测试数据生成方法、装置、介质及计算机系统。

背景技术

漏洞测试，是指对软件应用、系统、程序等进行检测，以发现其中可能存在的安全漏洞或缺陷的过程。为了提高测试中对于软件程序的运行逻辑的各个分支的覆盖率，从而提高对于程序漏洞的发掘效率，通常需要大量的测试数据。

发明内容

本说明书实施例提供一种测试数据生成方法、装置、存储介质及计算机系统，以部分解决上述现有技术存在的问题。

本说明书提供了一种测试数据生成方法，包括：获取触发目标程序异常的第一测试数据，提取第一测试数据对应的第一数据特征；运行第一测试数据对应的测试用例，得到第一运行报告，提取第一运行报告的第一报告特征；通过更改第一测试数据中包含的字段的值，得到多个第二测试数据，提取多个第二测试数据对应的多个第二数据特征；根据第一数据特征分别与各个第二数据特征的第一相似度，从多个第二数据特征中确定出第三数据特征，所述第三数据特征对应于多个第二测试数据中的第三测试数据；运行第三测试数据对应的测试用例，得到第二运行报告，提取第二运行报告对应的第二报告特征；以第二报告特征作为初始值，以第一报告特征为目标进行特征变异，得到变异报告特征；确定变异报告特征对应的第四测试数据，所述第三测试数据、第四测试数据用于测试目标程序。

本说明书提供了一种计算机系统，包括存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序，其特征在于，所述处理器执行所

述程序时实现如下步骤：获取触发目标程序异常的第一测试数据，提取第一测试数据对应的第一数据特征；运行第一测试数据对应的测试用例，得到第一运行报告，提取第一运行报告的第一报告特征；通过更改第一测试数据中包含的字段的值，得到多个第二测试数据，提取多个第二测试数据对应的多个第二数据特征；根据第一数据特征分别与各个第二数据特征的第一相似度，从多个第二数据特征中确定出第三数据特征，所述第三数据特征对应于多个第二测试数据中的第三测试数据；运行第三测试数据对应的测试用例，得到第二运行报告，提取第二运行报告对应的第二报告特征；以第二报告特征作为初始值，以第一报告特征为目标进行特征变异，得到变异报告特征；确定变异报告特征对应的第四测试数据，所述第三测试数据、第四测试数据用于测试目标程序。

本说明书提供了一种测试数据生成装置，所述装置包括：特征获取模块，用于获取触发目标程序异常的第一测试数据，提取第一测试数据对应的第一数据特征；运行第一测试数据对应的测试用例，得到第一运行报告，提取第一运行报告的第一报告特征；第一生成模块，用于通过更改第一测试数据中包含的字段的值，得到多个第二测试数据，提取多个第二测试数据对应的多个第二数据特征；根据第一数据特征分别与各个第二数据特征的第一相似度，从多个第二数据特征中确定出第三数据特征，所述第三数据特征对应于多个第二测试数据中的第三测试数据；第二生成模块，用于运行第三测试数据对应的测试用例，得到第二运行报告，提取第二运行报告对应的第二报告特征；以第二报告特征作为初始值，以第一报告特征为目标进行特征变异，得到变异报告特征；确定变异报告特征对应的第四测试数据，所述第三测试数据、第四测试数据用于测试目标程序。

本说明书提供了一种计算机可读非易失性存储介质，所述存储介质存储有计算机程序，所述计算机程序被处理器执行时实现上述的测试数据生成方法。

本说明书实施例采用的上述至少一个技术方案能够达到以下有益效果中的至少一项：能够自动生成大量高质量的针对测试目标程序的测试数据，提高对于目标程序执行逻辑分支的测试覆盖率，提高对于程序漏洞的发掘效率。

附图说明

此处所说明的附图用来提供对本说明书的进一步理解，构成本说明书的一部分，本说明书的示意性实施例及其说明用于解释本说明书，并不构成对本说明书的不当限定。在附图中：

图 1 为本说明书中一种测试数据生成方法的示意图；

图 2 为本说明书中一种测试数据生成方法的流程示意图；

图 3 为本说明书中特征变异的示意图；

图 4 为本说明书中通过神经网络生成报告特征对应的数据特征的示意图；

图5为本说明书中根据报告特征生成数据特征的神经网络的训练方式的示意图；

图 6 为本说明书中生成融合测试数据的示意图；

图 7 为本说明书中一种测试数据生成装置的结构示意图；

图 8 为本说明书实施例提供的计算机系统的结构示意图。

具体实施方式

为使本说明书的目的、技术方案和优点更加清楚，下面将结合本说明书具体实施例及相应的附图对本说明书技术方案进行清楚、完整地描述。显然，所描述的实施例仅是本说明书一部分实施例，而不是全部的实施例。基于本说明书中的实施例，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，都属于本说明书保护的范围。

如前所述，漏洞测试，是指对软件应用、系统、程序等进行检测，以发现其中可能存在的安全漏洞或缺陷的过程。为了提高测试中对于软件程序的运行逻辑的各个分支的覆盖率，从而提高对于程序漏洞的发掘效率，通常需要大量的测试数据。目前，一种获取测试数据的方式是通过人工构建测试数据。但是，仅仅通过人工方式构建测试数据，不仅需要消耗大量的人工工作量，而且通过人工构建的测试数据常常受到构建人员的经验或思维方式的限制，导致难以构建出对于被测程序的逻辑分支高覆盖率的测试数据。另一种获取测试数据的方式，是根据预设的更改规则（例如更改测试数据中部分字段的值）对已有的程序漏洞测试数据进行更改，从而生成新的测试数据。但是，软件漏洞往往隐藏在复杂的代码结构中，根据预设规则修改测试数据，往往只是在数据模式上浅

层修改，通过这种方式修改的测试数据，常常难以有效识别和预测复杂的代码结构中潜在的软件漏洞。因此，无论是单纯通过人工方式构建测试数据，还是通过预设规则修改已有测试数据获取测试数据，均难以得到大量且高覆盖率的测试数据，导致难以高效率的发掘被测软件中的漏洞。

为了解决上述问题，在本申请中提供一种测试数据生成方法。图 1 为本说明书中一种测试数据生成方法的示意图。如图 1 所示，可以获取已触发过目标程序的异常的测试数据 D1,提取 D1 的特征（例如为 A1），并运行测试数据 D1 对应的测试用例，得到运行报告 R1，提取 R1 的特征（例如为 RA1）；变更测试数据 D1 中一到多个字段的值，例如得到测试数据 D2，运行 D2 对应的测试用例，得到运行报告 R2，提取 R2 的特征（例如为 RA2）；通过神经网络，以 RA2 为 R 初始值，以 RA1 为目标进行特征变异，得到变异报告特征 RA3；根据 RA3 得到数据特征 A3，根据数据特征 A2 和数据特征 A3 分别与数据特征 A1 的相似度，确定是否将数据特征 A2 和数据特征 A3 用于测试目标程序。

本申请中提供的测试数据生成方法，通过字段更改得到扩展的测试数据。并通过基于深度学习的、对于扩展测试数据的运行报告特征的变异，获取变异报告特征，并通过报告变异特征获取进一步扩展的测试数据。因此，第一，通过该方法能够通过能够自动生成测试数据，减少了对于人工生成测试数据的依赖，提高了测试数据获取的效率。第二，通过运行报告特征的变异，常常可以获取超出人类常规识别的关键特征，并基于这些关键特征生成高质量的测试数据。第三，由于运行报告中通常包括测试用例的直接运行结果、运行错误或抛出异常的描述，通过运行报告变异获取的测试数据相对于单纯通过字段值变化得到的测试数据，常常可以更为有效的扩大对于程序中逻辑分支的覆盖率，从而更有效的识别潜在的软件漏洞。

以下结合附图，详细说明本说明书各实施例提供的技术方案。

图 2 为本说明书中一种测试数据生成方法的流程示意图，具体可包括以下步骤：

S200：获取触发目标程序异常的第一测试数据，提取第一测试数据对应的第一数据特征；运行第一测试数据对应的测试用例，得到第一运行报告，提取第一运行报告的第一报告特征。在不同的实施例中，目标程序可以是用于不同

具体业务、或用于不同具体目的软件程序，本说明书对此不做限制。

测试数据，或称为测试用例数据，是用于对软件程序或系统执行测试用例的一组特定数据。通常，不同的测试数据可以是根据不同测试用例的测试目标而确定，旨在覆盖软件运行中各种可能的执行分支和边界条件。因此，在不同的实施例中，测试用例数据可以包括程序输入数据的例如正常值、异常值、边界值、或无效的等不同类型的数据。进而，通过高质量的测试数据，可以有效地发现软件中的漏洞。

在不同的实施例中，测试数据所具有的数据格式可以不同。在一个实施例中，例如可以是 Yaml（JavaScript Object Notation）格式。通常，测试数据中可以包括多个数据字段，在不同的具体实施例中，测试数据包括的具体数据字段可以不同。例如，在一个实施例中，可以包括针对目标程序的多种输入数据。在另一个实施例中，还可以包括测试用例的执行条件或预期结果中的一种或多种。

在不同的实施例中，提取测试数据对应的数据特征的方式可以不同。在一个实施例中，可以将根据测试数据，生成对应的多维向量特征，多维向量特征中的各个维度的分量可以根据测试数据中各个字段的名称或字段值确定。

在获取第一测试数据之后，可以运行第一测试数据对应的测试用例，得到第一运行报告，提取第一运行报告的第一报告特征。测试用例的运行报告，是对测试用例的执行情况的记录。在不同的实施例中，运行报告包括的具体内容可以不同。在一个实施例中，例如可以包括测试结果（例如通过、失败或阻塞中的一种）、执行时间、错误描述（例如，若测试失败，对于错误情况的描述或对于程序抛出异常的描述）、漏洞编号、测试日志和测试环境信息中的一种或多种。

在不同的实施例中，提取运行报告的报告特征的具体方式可以不同，在一个实施例中，可以通过神经网络提取运行报告的报告特征。

S202：通过更改第一测试数据中包含的字段的值，得到多个第二测试数据，提取多个第二测试数据对应的多个第二数据特征；根据第一数据特征分别与各个第二数据特征的第一相似度，从多个第二数据特征中确定出第三数据特征，所述第三数据特征对应于多个第二测试数据中的第三测试数据。

具体的，在一个实施例中，可以通过多次更改第一测试数据中包括的至少部分字段的值，得到多个第二测试数据。该步骤中，提取第二数据特征的方式，可以类似于步骤S200中提取测试数据对应的数据特征的方式，这里不再赘述。

在不同的实施例中，确定第一数据特征与各个第二数据特征的第一相似度的具体方式可以不同。在一个实施例中，可以通过计算第一数据特征与各个第二数据特征的余弦相似度，确定第一数据特征与各个第二数据特征之间的第一相似度。在不同的实施例中，从多个第二数据特征中确定出第三数据特征的具体方式也可以不同。在一个实施例中，可以根据第一数据特征分别与多个第二数据特征的相似度（第一相似度）的值，对多个第二数据特征进行排序，将排序后位置居前的例如第一预定比例的第二数据特征，确定为第三数据特征。在不同的具体实施例中，第一预定比例的取值可以不同。在一个具体的实施例中，例如可以将排序后位置居前的例如10%的第二数据特征，确定为第三数据特征。

S204：运行第三测试数据对应的测试用例，得到第二运行报告，提取第二运行报告对应的第二报告特征；以第二报告特征作为初始值，以第一报告特征为目标进行特征变异，得到变异报告特征；确定变异报告特征对应的第四测试数据，所述第三测试数据、第四测试数据用于测试目标程序。该步骤中，提取第二报告特征的方式，可以类似于 S200 中提取运行报告对应的报告特征的方式，这里不再赘述。

在不同的实施例中，特征变异的具体方式可以不同。在一个实施例中，可以基于快速梯度下降算法，以第二报告特征作为初始值，以第一报告特征为目标进行特征变异，得到变异报告特征。具体的，图 3 为本说明书中特征变异的示意图。如图 3 所示，第一报告特征例如可以表示为向量$\vec{V}$，第二报告特征例如可以表示为向量$\overrightarrow{V'}$。可以将$\overrightarrow{V'}$和$\vec{V}$输入神经网络，通过快速梯度下降算法，以$\overrightarrow{V'}$作为特征变异的初始值，使得$\overrightarrow{V'}$向逼近$\vec{V}$的方向进行变异，得到新的变异报告特征（表示为$\overrightarrow{V^*}$）。

图 4 为本说明书中通过神经网络生成报告特征对应的数据特征的示意图。如图 4 所示，在一个实施例中，可以将变异报告特征（报告特征RA3）输入预先训练的第一神经网络（神经网络 N2），得到第四数据特征（数据特征 A3），所述第一神经网络用于根据运行报告的特征，输出测试数据的特征；根据第四

数据特征，确定所述第四测试数据。图 5 为本说明书中根据报告特征生成数据特征的神经网络的训练方式的示意图。如图 5 所示，在一个具体的实施例中，在所述第一神经网络的预先训练中，可以以所述第一报告特征（报告特征 RA1）、第二报告特征（报告特征 RA2）作为训练样本，以第一数据特征（数据特征 A1）、第二数据特征（数据特征 A2）作为训练样本对应的训练标签。在一个实施例中，可以根据第四数据特征对应的多维向量中各个维度的分量的值，确定所述第四测试数据中各个字段的名称或字段。

为了进一步提高用于测试目标程序的测试数据的质量，在一个实施例中，可以确定第一数据特征和变异数据特征之间的第三相似度，若所述第三相似度达到预定阈值，则所述第四测试数据用于测试目标程序。在不同的具体实施例中，预定阈值的具体取值可以不同。

为了使进一步扩展测试数据的数量，在一个实施例中，还可以从第三测试数据和第四测试数据中确定出优选测试数据和次选测试数据，结合所述优选测试数据和次选测试数据，得到第五测试数据，所述第五测试数据用于测试目标程序。在不同的实施例中，确定优选测试数据和次选测试数据的方式可以不同。在一个实施例中，例如可以将第三测试数据和第四测试数据中第二预定比例的数量的测试数据作为优选测试数据，其余的测试数据作为次选测试数据。各个优选测试数据的数据特征与第一数据特征的相似度，高于各个次选测试的数据特征与第一数据特征的相似度。在不同的具体实施例中，第二预定比例的值可以不同。在一个具体的实施例中，例如可以将第三测试数据和第四测试数据中例如 20%数量的测试数据作为优选测试数据，其余的测试数据作为次选测试数据。

在不同的实施例中，结合所述优选测试数据和次选测试数据的具体方式可以不同，在一个具体的实施例中，可以根据优选测试数据中第一字段的值、以及次选测试数据中第二字段的值，确定第五测试数据中第一字段和第二字段的值。图 6 为本说明书中生成融合测试数据的示意图。如图 6 所示，测试数据的字段例如包括字段 C1、字段 C2、字段 C3 和字段 C4。其中，第一字段可以包括字段 C1 和字段 C2，第二字段可以包括字段 C3 和字段 C4。可以根据优选测试数据 D3 中字段 C1 和字段 C2 的值、以及次选测试数据 D4 中字段 C3 和字

段 C4 的值，确定融合测试数据（第五测试数据）D5 中字段 C1、字段 C2、字段 C3 和字段 C4 的值。

类似的，为了进一步提高用于测试目标程序的测试数据的质量，在另一个具体的实施例中，确定第五测试数据对应的数据特征与第一数据特征之间的第二相似度，若所述第二相似度达到预定阈值，则所述第五测试数据用于测试目标程序。

以上为本说明书实施例提供的测试数据生成方法，基于同样的思路，本说明书还提供了相应的测试数据生成装置。如图 7 所示，该测试数据生成装置包括：特征获取模块 702，用于获取触发目标程序异常的第一测试数据，提取第一测试数据对应的第一数据特征；运行第一测试数据对应的测试用例，得到第一运行报告，提取第一运行报告的第一报告特征；第一生成模块 704，用于通过更改第一测试数据中包含的字段的值，得到多个第二测试数据，提取多个第二测试数据对应的多个第二数据特征；根据第一数据特征分别与各个第二数据特征的第一相似度，从多个第二数据特征中确定出第三数据特征，所述第三数据特征对应于多个第二测试数据中的第三测试数据；第二生成模块 706，用于运行第三测试数据对应的测试用例，得到第二运行报告，提取第二运行报告对应的第二报告特征；以第二报告特征作为初始值，以第一报告特征为目标进行特征变异，得到变异报告特征；确定变异报告特征对应的第四测试数据，所述第三测试数据、第四测试数据用于测试目标程序。

为了描述的方便，描述以上装置时以功能分为各种模块分别描述。当然，在实施本说明书时可以把各模块的功能在同一个或多个软件和/或硬件中实现。

The specification also provides a computer-readable non-transitory storage medium, wherein the storage medium stores a computer program, which, when executed by a processor, can be used to perform one or more steps of one or more methods described or illustrated herein or provides functionality described or illustrated herein. Herein, the computer-readable non-transitory storage medium or media may include one or more semiconductor-based or other integrated circuits (ICs) (such, as for example, field-programmable gate arrays (FPGAs) or application-specific ICs (ASICs)), hard disk drives (HDDs), hybrid hard drives (HHDs), optical discs, optical disc drives (ODDs), magneto-optical discs, magneto-optical drives, floppy diskettes, floppy disk drives

(FDDs), magnetic tapes, solid-state drives (SSDs), RAM-drives, SECURE DIGITAL cards or drives, any other suitable computer-readable non-transitory storage media, or any suitable combination of two or more of these, where appropriate. A computer-readable non-transitory storage medium may be volatile, non-volatile, or a combination of volatile and non-volatile, where appropriate.

The specification also provides a computer system. FIG. 8 illustrates an example computer system 800. In particular embodiments, one or more computer systems 800 perform one or more steps of one or more methods described or illustrated herein. In particular embodiments, one or more computer systems 800 provide functionality described or illustrated herein. In particular embodiments, software running on one or more computer systems 800 performs one or more steps of one or more methods described or illustrated herein or provides functionality described or illustrated herein. Particular embodiments include one or more portions of one or more computer systems 800. Herein, reference to a computer system may encompass a computing device, and vice versa, where appropriate. Moreover, reference to a computer system may encompass one or more computer systems, where appropriate.

This disclosure contemplates any suitable number of computer systems 800. This disclosure contemplates computer system 800 taking any suitable physical form. As example and not by way of limitation, computer system 800 may be an embedded computer system, a system-on-chip (SOC), a single-board computer system (SBC) (such as, for example, a computer-on-module (COM) or system-on-module (SOM)), a desktop computer system, a laptop or notebook computer system, an interactive kiosk, a mainframe, a mesh of computer systems, a mobile telephone, a personal digital assistant (PDA), a server, a tablet computer system, or a combination of two or more of these. Where appropriate, computer system 800 may include one or more computer systems 800; be unitary or distributed; span multiple locations; span multiple machines; span multiple data centers; or reside in a cloud, which may include one or more cloud components in one or more networks. Where appropriate, one or more computer systems 800 may perform without substantial spatial or temporal limitation one or more steps of one or more methods described or illustrated herein. As an example and not by way of limitation, one or more computer systems 800 may perform in real time or in batch mode one or more steps of one or more methods

described or illustrated herein. One or more computer systems 800 may perform at different times or at different locations one or more steps of one or more methods described or illustrated herein, where appropriate.

In particular embodiments, computer system 800 includes a processor 802, memory 804, storage 806, an input/output (I/O) interface 808, a communication interface 810, and a bus 812. Although this disclosure describes and illustrates a particular computer system having a particular number of particular components in a particular arrangement, this disclosure contemplates any suitable computer system having any suitable number of any suitable components in any suitable arrangement.

In particular embodiments, processor 802 includes hardware for executing instructions, such as those making up a computer program. As an example and not by way of limitation, to execute instructions, processor 802 may retrieve (or fetch) the instructions from an internal register, an internal cache, memory 804, or storage 806; decode and execute them; and then write one or more results to an internal register, an internal cache, memory 804, or storage 806. In particular embodiments, processor 802 may include one or more internal caches for data, instructions, or addresses. This disclosure contemplates processor 802 including any suitable number of any suitable internal caches, where appropriate. As an example and not by way of limitation, processor 802 may include one or more instruction caches, one or more data caches, and one or more translation lookaside buffers (TLB s). Instructions in the instruction caches may be copies of instructions in memory 804 or storage 806, and the instruction caches may speed up retrieval of those instructions by processor 802. Data in the data caches may be copies of data of 80 in memory 804 or storage 806 for instructions executing at processor 802 to operate on; the results of previous instructions executed at processor 802 for access by subsequent instructions executing at processor 802 or for writing to memory 804 or storage 806; or other suitable data. The data caches may speed up read or write operations by processor 802. The TLBs may speed up virtual-address translation for processor 802. In particular embodiments, processor 802 may include one or more internal registers for data, instructions, or addresses. This disclosure contemplates processor 802 including any suitable number of any suitable internal registers, where appropriate. Where appropriate, processor 802 may include one or more arithmetic logic

units (ALUs); be a multi-core processor; or include one or more processors 802. Although this disclosure describes and illustrates a particular processor, this disclosure contemplates any suitable processor.

In particular embodiments, memory 804 includes main memory for storing instructions for processor 802 to execute or data for processor 802 to operate on. As an example and not by way of limitation, computer system 800 may load instructions from storage 806 or another source (such as, for example, another computer system 800) to memory 804. Processor 802 may then load the instructions from memory 804 to an internal register or internal cache. To execute the instructions, processor 802 may retrieve the instructions from the internal register or internal cache and decode them. During or after execution of the instructions, processor 802 may write one or more results (which may be intermediate or final results) to the internal register or internal cache. Processor 802 may then write one or more of those results to memory 804. In particular embodiments, processor 802 executes only instructions in one or more internal registers or internal caches or in memory 804 (as opposed to storage 806 or elsewhere) and operates only on data in one or more internal registers or internal caches or in memory 804 (as opposed to storage 806 or elsewhere). One or more memory buses (which may each include an address bus and a data bus) may couple processor 802 to memory 804. Bus 812 may include one or more memory buses, as described below. In particular embodiments, one or more memory management units (MMUs) reside between processor 802 and memory 804 and facilitate accesses to memory 804 requested by processor 802. In particular embodiments, memory 804 includes random access memory (RAM). This RAM may be volatile memory, where appropriate. Where appropriate, this RAM may be dynamic RAM (DRAM) or static RAM (SRAM). Moreover, where appropriate, this RAM may be single-ported or multi-ported RAM. This disclosure contemplates any suitable RAM. Memory 804 may include one or more memories 804, where appropriate. Although this disclosure describes and illustrates particular memory, this disclosure contemplates any suitable memory.

In particular embodiments, storage 806 includes mass storage for data or instructions. As an example and not by way of limitation, storage 806 may include a hard disk drive (HDD), a floppy disk drive, flash memory, an optical disc, a magneto-optical disc, magnetic tape, or a Universal Serial Bus (USB) drive or a combination of two or more of these. Storage 806 may include

removable or non-removable (or fixed) media, where appropriate. Storage 806 may be internal or external to computer system 800, where appropriate. In particular embodiments, storage 806 is non-volatile, solid-state memory. In particular embodiments, storage 806 includes read-only memory (ROM). Where appropriate, this ROM may be mask-programmed ROM, programmable ROM (PROM), erasable PROM (EPROM), electrically erasable PROM (EEPROM), electrically alterable ROM (EAROM), or flash memory or a combination of two or more of these. This disclosure contemplates mass storage 806 taking any suitable physical form. Storage 806 may include one or more storage control units facilitating communication between processor 802 and storage 806, where appropriate. Where appropriate, storage 806 may include one or more storages 806. Although this disclosure describes and illustrates particular storage, this disclosure contemplates any suitable storage.

In particular embodiments, I/O interface 808 includes hardware, software, or both, providing one or more interfaces for communication between computer system 800 and one or more I/O devices. Computer system 800 may include one or more of these I/O devices, where appropriate. One or more of these I/O devices may enable communication between a person and computer system 800. As an example and not by way of limitation, an I/O device may include a keyboard, keypad, microphone, monitor, mouse, printer, scanner, speaker, still camera, stylus, tablet, touch screen, trackball, video camera, another suitable I/O device or a combination of two or more of these. An I/O device may include one or more sensors. This disclosure contemplates any suitable I/O devices and any suitable I/O interfaces 808 for them. Where appropriate, I/O interface 808 may include one or more device or software drivers enabling processor 802 to drive one or more of these I/O devices. I/O interface 808 may include one or more I/O interfaces 808, where appropriate. Although this disclosure describes and illustrates a particular I/O interface, this disclosure contemplates any suitable I/O interface.

In particular embodiments, communication interface 810 includes hardware, software, or both providing one or more interfaces for communication (such as, for example, packet-based communication) between computer system 800 and one or more other computer systems 800 or one or more networks. As an example and not by way of limitation, communication interface 810 may include a network interface controller (NIC) or network adapter for communicating with an

Ethernet or other wire-based network or a wireless NIC (WNIC) or wireless adapter for communicating with a wireless network, such as a WI-FI network. This disclosure contemplates any suitable network and any suitable communication interface 810 for it. As an example and not by way of limitation, computer system 800 may communicate with an ad hoc network, a personal area network (PAN), a local area network (LAN), a wide area network (WAN), a metropolitan area network (MAN), or one or more portions of the Internet or a combination of two or more of these. One or more portions of one or more of these networks may be wired or wireless. As an example, computer system 800 may communicate with a wireless PAN (WPAN) (such as, for example, a BLUETOOTH WPAN), a WI-FI network, a WI-MAX network, a cellular telephone network (such as, for example, a Global System for Mobile Communications (GSM) network), or other suitable wireless network or a combination of two or more of these. Computer system 800 may include any suitable communication interface 810 for any of these networks, where appropriate. Communication interface 810 may include one or more communication interfaces 810, where appropriate. Although this disclosure describes and illustrates a particular communication interface, this disclosure contemplates any suitable communication interface.

In particular embodiments, bus 812 includes hardware, software, or both coupling components of computer system 800 to each other. As an example and not by way of limitation, bus 812 may include an Accelerated Graphics Port (AGP) or other graphics bus, an Enhanced Industry Standard Architecture (EISA) bus, a front-side bus (FSB), a HYPERTRANSPORT (HT) interconnect, an Industry Standard Architecture (ISA) bus, an INFINIBAND interconnect, a low-pin-count (LPC) bus, a memory bus, a Micro Channel Architecture (MCA) bus, a Peripheral Component Interconnect (PCI) bus, a PCI-Express (PCIe) bus, a serial advanced technology attachment (SATA) bus, a Video Electronics Standards Association local (VLB) bus, or another suitable bus or a combination of two or more of these. Bus 812 may include one or more buses 812, where appropriate. Although this disclosure describes and illustrates a particular bus, this disclosure contemplates any suitable bus or interconnect.

还需要说明的是，术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含，从而使得包括一系列要素的过程、方法、商品或者设备不仅包括那些要素，而且还包括没有明确列出的其他要素，或者是还包括为这种过程、

方法、商品或者设备所固有的要素。在没有更多限制的情况下，由语句“包括一个……”限定的要素，并不排除在包括所述要素的过程、方法、商品或者设备中还存在另外的相同要素。

本说明书中的各个实施例均采用递进的方式描述，各个实施例之间相同相似的部分互相参见即可，每个实施例重点说明的都是与其他实施例的不同之处。尤其，对于系统实施例而言，由于其基本相似于方法实施例，所以描述的比较简单，相关之处参见方法实施例的部分说明即可。

以上所述仅为本说明书的实施例而已，并不用于限制本说明书。对于本领域技术人员来说，本说明书可以有各种更改和变化。凡在本说明书的精神和原理之内所作的任何修改、等同替换、改进等，均应包含在本申请的权利要求范围之内。

权利要求书

1、一种测试数据生成方法，包括：

获取触发目标程序异常的第一测试数据，提取第一测试数据对应的第一数据特征；运行第一测试数据对应的测试用例，得到第一运行报告，提取第一运行报告的第一报告特征；

通过更改第一测试数据中包含的字段的值，得到多个第二测试数据，提取多个第二测试数据对应的多个第二数据特征；根据第一数据特征分别与各个第二数据特征的第一相似度，从多个第二数据特征中确定出第三数据特征，所述第三数据特征对应于多个第二测试数据中的第三测试数据；

运行第三测试数据对应的测试用例，得到第二运行报告，提取第二运行报告对应的第二报告特征；以第二报告特征作为初始值，以第一报告特征为目标进行特征变异，得到变异报告特征；确定变异报告特征对应的第四测试数据，所述第三测试数据、第四测试数据用于测试目标程序。

2、根据权利要求 1 所述的方法，还包括：

从第三测试数据和第四测试数据中确定出优选测试数据和次选测试数据，结合所述优选测试数据和次选测试数据，得到第五测试数据，所述第五测试数据用于测试目标程序。

3、根据权利要求 2 所述的方法，其中，结合所述优选测试数据和次选测试数据，得到第五测试数据，包括：

根据优选测试数据中第一字段的值、以及次选测试数据中第二字段的值，确定第五测试数据中第一字段和第二字段的值。

4、根据权利要求 2 所述的方法，其中，所述第五测试数据用于测试目标程序，包括：

确定第五测试数据对应的数据特征与第一数据特征之间的第二相似度，若所述第二相似度达到预定阈值，则所述第五测试数据用于测试目标程序。

5、根据权利要求 1 所述的方法，其中，确定变异报告特征对应的第四测

试数据，包括：

将变异报告特征输入预先训练的第一神经网络，得到第四数据特征，所述第一神经网络用于根据运行报告的特征，输出测试数据的特征；

根据第四数据特征，确定所述第四测试数据。

6、根据权利要求 5 所述的方法，其中，在所述第一神经网络的预先训练中，以所述第一报告特征、第二报告特征作为训练样本，以第一数据特征、第三数据特征作为训练样本对应的训练标签。

7、根据权利要求 1 所述的方法，其中，以第二报告特征作为初始值，以第一报告特征为目标进行特征变异，得到变异报告特征，包括：

基于快速梯度下降算法，以第二报告特征作为初始值，以第一报告特征为目标进行特征变异，得到变异报告特征。

8、根据权利要求 1 所述的方法，其中，第四测试数据用于测试目标程序，包括：

确定第一数据特征和变异数据特征之间的第三相似度，若所述第三相似度达到预定阈值，则所述第四测试数据用于测试目标程序。

9、一种计算机系统，包括存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序，其特征在于，所述处理器执行所述程序时实现如下步骤：

获取触发目标程序异常的第一测试数据，提取第一测试数据对应的第一数据特征；运行第一测试数据对应的测试用例，得到第一运行报告，提取第一运行报告的第一报告特征；

通过更改第一测试数据中包含的字段的值，得到多个第二测试数据，提取多个第二测试数据对应的多个第二数据特征；根据第一数据特征分别与各个第二数据特征的第一相似度，从多个第二数据特征中确定出第三数据特征，所述第三数据特征对应于多个第二测试数据中的第三测试数据；

运行第三测试数据对应的测试用例，得到第二运行报告，提取第二运行报

告对应的第二报告特征；以第二报告特征作为初始值，以第一报告特征为目标进行特征变异，得到变异报告特征；确定变异报告特征对应的第四测试数据，所述第三测试数据、第四测试数据用于测试目标程序。

10、如权利要求 9 所述的计算机系统，其中，所述步骤还包括：

从第三测试数据和第四测试数据中确定出优选测试数据和次选测试数据，结合所述优选测试数据和次选测试数据，得到第五测试数据，所述第五测试数据用于测试目标程序。

11、如权利要求 10 所述的计算机系统，其中，结合所述优选测试数据和次选测试数据，得到第五测试数据，包括：

根据优选测试数据中第一字段的值、以及次选测试数据中第二字段的值，确定第五测试数据中第一字段和第二字段的值。

12、如权利要求 10 所述的计算机系统，其中，所述第五测试数据用于测试目标程序，包括：

确定第五测试数据对应的数据特征与第一数据特征之间的第二相似度，若所述第二相似度达到预定阈值，则所述第五测试数据用于测试目标程序。

13、如权利要求 9 所述的计算机系统，其中，确定变异报告特征对应的第四测试数据，包括：

将变异报告特征输入预先训练的第一神经网络，得到第四数据特征，所述第一神经网络用于根据运行报告的特征，输出测试数据的特征；

根据第四数据特征，确定所述第四测试数据。

14、如权利要求 13 所述的计算机系统，其中，在所述第一神经网络的预先训练中，以所述第一报告特征、第二报告特征作为训练样本，以第一数据特征、第三数据特征作为训练样本对应的训练标签。

15、如权利要求 9 所述的计算机系统，其中，以第二报告特征作为初始值，以第一报告特征为目标进行特征变异，得到变异报告特征，包括：

基于快速梯度下降算法，以第二报告特征作为初始值，以第一报告特征为

目标进行特征变异，得到变异报告特征。

16、如权利要求 9 所述的计算机系统，其中，第四测试数据用于测试目标程序，包括：

确定第一数据特征和变异数据特征之间的第三相似度，若所述第三相似度达到预定阈值，则所述第四测试数据用于测试目标程序。

17、一种测试数据生成装置，所述装置包括：

特征获取模块，用于获取触发目标程序异常的第一测试数据，提取第一测试数据对应的第一数据特征；运行第一测试数据对应的测试用例，得到第一运行报告，提取第一运行报告的第一报告特征；

第一生成模块，用于通过更改第一测试数据中包含的字段的值，得到多个第二测试数据，提取多个第二测试数据对应的多个第二数据特征；根据第一数据特征分别与各个第二数据特征的第一相似度，从多个第二数据特征中确定出第三数据特征，所述第三数据特征对应于多个第二测试数据中的第三测试数据；

第二生成模块，用于运行第三测试数据对应的测试用例，得到第二运行报告，提取第二运行报告对应的第二报告特征；以第二报告特征作为初始值，以第一报告特征为目标进行特征变异，得到变异报告特征；确定变异报告特征对应的第四测试数据，所述第三测试数据、第四测试数据用于测试目标程序。

18、一种计算机可读非易失性存储介质，其特征在于，所述存储介质存储有计算机程序，所述计算机程序被处理器执行时实现上述权利要求 1-8 任一项所述的方法。