



(12) 发明专利申请

(10) 申请公布号 CN 112416662 A

(43) 申请公布日 2021.02.26

(21) 申请号 202011349264.X

(22) 申请日 2020.11.26

(71) 申请人 清华大学

地址 100084 北京市海淀区清华园

(72) 发明人 裴丹 苏亚

(74) 专利代理机构 北京清亦华知识产权代理事

务所(普通合伙) 11201

代理人 韩海花

(51) Int. Cl.

G06F 11/16 (2006.01)

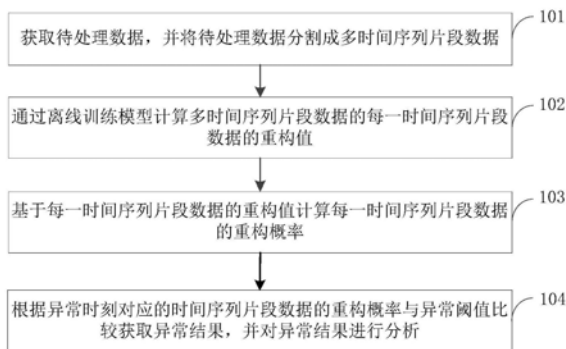
权利要求书2页 说明书11页 附图3页

(54) 发明名称

多时间序列数据异常检测方法与装置

(57) 摘要

本申请提出一种多时间序列数据异常检测方法... 获取待处理数据, 并将待处理数据分割成多时间序列片段数据; 通过离线训练模型计算多时间序列片段数据的每一时间序列片段数据的重构值; 基于每一时间序列片段数据的重构值计算每一时间序列片段数据的重构概率; 根据异常时刻对应的时间序列片段数据的重构概率与异常阈值比较获取异常结果, 并对异常结果进行分析。由此, 在考虑多时间序列随机性和时间依赖性的前提下, 对多时间序列的历史正常模式进行学习, 从而实现对多时间序列数据更加准确的多时间序列数据异常检测, 并使模型的输出结果具有可解释性。



1. 一种多时间序列数据异常检测方法,其特征在于,所述方法包括:
获取待处理数据,并将所述待处理数据分割成多时间序列片段数据;
通过离线训练模型计算所述多时间序列片段数据的每一时间序列片段数据的重构值;
基于所述每一时间序列片段数据的重构值计算每一时间序列片段数据的重构概率;
根据异常时刻对应的时间序列片段数据的重构概率与异常阈值比较获取异常结果,并对所述异常结果进行分析。

2. 如权利要求1所述的多时间序列数据异常检测方法,其特征在于,所述获取待处理数据,将所述待处理数据分割成多时间序列片段数据,包括:

获取历史数据,将所述历史数据中的缺失数据和错误数据统一设置为零,得到清洗后的历史数据;

对所述清洗后的历史数据进行基于均值方差的标准化处理得到所述待处理数据;

通过滑动窗口将所述待处理数据分割成所述多时间序列片段数据。

3. 如权利要求1所述的多时间序列数据异常检测方法,其特征在于,所述离线训练模型包括构建变分网络和构建生成网络;所述通过离线训练模型计算所述多时间序列片段数据的每一时间序列片段数据的重构值,包括:

所述变分网络将多时间序列片段数据的每一时间序列片段数据映射到随机隐变量;

所述生成网络通过随机隐变量连接技术连接多时间序列片段的随机隐变量获得每一时间序列片段数据的重构值。

4. 如权利要求3所述的多时间序列数据异常检测方法,其特征在于,所述变分网络将多时间序列片段数据的每一时间序列片段数据映射到随机隐变量,包括:

将 t 时刻的时间序列片段数据 x_t 和 $t-1$ 时刻的门控循环神经网络GRU隐藏变量 e_{t-1} 输入变分网络的门控循环神经网络单元,生成门控循环神经网络隐藏变量 e_t ;

将所述门控循环神经网络隐藏变量 e_t 与 $t-1$ 时刻的随机隐变量 z_{t-1} 进行拼接;

将拼接后的 e_t 与 z_{t-1} 输入变分网络的全连接层、软加激活函数层和线性层获得 t 时刻的随机隐变量 z_t 的均值和标准差;

基于所述随机隐变量 z_t 的均值和标准差,采用平面标准化流拟合得到随机隐变量 z_t 。

5. 如权利要求4所述的多时间序列数据异常检测方法,其特征在于,所述生成网络通过随机隐变量连接技术连接多时间序列片段的随机隐变量获得每一时间序列片段数据的重构值,包括:

采用线性高斯状态空间模型连接相邻时刻的随机隐变量;

将 t 时刻的随机隐变量 z_t 和 $t-1$ 时刻的门控循环神经网络隐藏变量 d_{t-1} 输入生成网络的门控循环神经网络单元生成门控循环神经网络隐藏变量 d_t ;

将所述门控循环神经网络隐藏变量 d_t 输入生成网络的全连接层、软加激活函数层和线性层生成重构值的均值和标准差;

通过所述重构值的均值和标准差计算输入数据的重构值。

6. 如权利要求3所述的多时间序列数据异常检测方法,其特征在于,还包括:

采用最大化ELBO训练所述变分网络和所述生成网络并获取多时间序列片段数据的重构误差;

通过所述多时间序列片段数据的重构误差计算多时间序列片段数据的每一时间序列

片段数据的重构概率；

基于多时间序列片段数据的每一时间序列片段数据的重构概率，采用POT方法确定所述异常阈值。

7. 如权利要求1所述的多时间序列数据异常检测方法，其特征在于，所述对所述异常结果进行分析，包括：

将所述异常时刻对应的时间序列片段数据的重构概率分解到单时间序列；

对每个单时间序列的重构概率进行升序排列。

8. 一种多时间序列数据异常检测装置，包括：

获取分割模块，用于获取待处理数据，并将所述待处理数据分割成多时间序列片段数据；

第一计算模块，用于通过离线训练模型计算所述多时间序列片段数据的每一时间序列片段数据的重构值；

第二计算模块，基于所述每一时间序列片段数据的重构值计算每一时间序列片段数据的重构概率；

比较模块，用于在线计算异常时刻数据的重构概率并与所述异常阈值比较获取异常结果；

分析模块，对所述异常结果进行分析。

9. 一种电子设备，其特征在于，包括：

至少一个处理器；以及

与所述至少一个处理器耦合连接的存储器；其中，

所述存储器存储有计算机程序，所述计算机程序能够被所述至少一个处理器执行，以实现权利要求1-7任一项所述的方法。

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用于存储计算机程序，当所述计算机程序被执行时，能够实现权利要求1-7任一项所述的方法。

多时间序列数据异常检测方法与装置

技术领域

[0001] 本申请涉及互联网技术领域,尤其涉及一种多时间序列数据异常检测方法与装置。

背景技术

[0002] 互联网服务是当今互联网的一个重要内容,这些服务部署在互联网公司大量的机器(服务器、虚拟机、容器)上。为了确保这些服务的可靠性并为上层软件和系统提供更好的服务支撑,运维人员需要对机器的运行状态进行质量检测。机器是整个互联网的硬件基础,在运维管理工作中,运维工程师通常会监控采集机器的各个性能指标,这些来自同一机器的不同指标组成了多时间序列。例如一台机器有CPU使用率、网络使用率、内存使用率、磁盘使用率等指标。

[0003] 在实际使用过程中,由于机器老化、性能过载、恶意攻击等故障,机器性能不可避免的会受到影响,这些监控指标也会发生异常。多时间序列的异常检测是一个非常重要的问题,可以帮助运维工程师迅速发现机器的异常并及时止损。

发明内容

[0004] 本申请旨在至少在一定程度上解决相关技术中的技术问题之一。

[0005] 为此,本申请的一个目的在于提出一种多时间序列数据异常检测方法和装置,在考虑多时间序列随机性和时间依赖性的前提下,对多时间序列的历史正常模式进行学习,从而实现对多时间序列数据更加准确的多时间序列数据异常检测,并使模型的输出结果具有可解释性。

[0006] 本申请的另一个目的在于提出一种多时间序列数据异常检测装置。

[0007] 为达到上述目的,本申请实施例提出了一种多时间序列数据异常检测方法,包括以下步骤:

[0008] 获取待处理数据,并将所述待处理数据分割成多时间序列片段数据;

[0009] 通过离线训练模型计算所述多时间序列片段数据的每一时间序列片段数据的重构值;

[0010] 基于所述每一时间序列片段数据的重构值计算每一时间序列片段数据的重构概率;

[0011] 根据异常时刻对应的时间序列片段数据的重构概率与异常阈值比较获取异常结果,并对所述异常结果进行分析。

[0012] 本申请实施例的多时间序列数据异常检测方法,获取待处理数据,并将待处理数据分割成多时间序列片段数据;通过离线训练模型计算多时间序列片段数据的每一时间序列片段数据的重构值;基于每一时间序列片段数据的重构值计算每一时间序列片段数据的重构概率;根据异常时刻对应的时间序列片段数据的重构概率与异常阈值比较获取异常结果,并对异常结果进行分析。由此,在考虑多时间序列随机性和时间依赖性的前提下,对多

时间序列的历史正常模式进行学习,从而实现了对多时间序列数据更加准确的多时间序列数据异常检测,并使模型的输出结果具有可解释性。

[0013] 另外,根据本申请上述实施例的多时间序列数据异常检测方法,还可以具有如下附加的技术特征:

[0014] 进一步地,在本申请的一个实施例中,所述获取待处理数据,将所述待处理数据分割成多时间序列片段数据,包括:

[0015] 获取历史数据,将所述历史数据中的缺失数据和错误数据统一设置为零,得到清洗后的历史数据;

[0016] 对所述清洗后的历史数据进行基于均值方差的标准化处理得到所述待处理数据;

[0017] 通过滑动窗口将所述待处理数据分割成所述多时间序列片段数据。

[0018] 进一步地,在本申请的一个实施例中,所述离线训练模型包括构建变分网络和构建生成网络。

[0019] 进一步地,在本申请的一个实施例中,所述通过离线训练模型计算所述多时间序列片段数据的每一时间序列片段数据的重构值,包括:

[0020] 所述变分网络将多时间序列片段数据的每一时间序列片段数据映射到随机隐变量;

[0021] 所述生成网络通过随机隐变量连接技术连接多时间序列片段的随机隐变量获得每一时间序列片段数据的重构值。

[0022] 进一步地,在本申请的一个实施例中,所述变分网络将多时间序列片段数据的每一时间序列片段数据映射到随机隐变量,包括:

[0023] 将 t 时刻的时间序列片段数据 x_t 和 $t-1$ 时刻的门控循环神经网络GRU隐藏变量 e_{t-1} 输入变分网络的门控循环神经网络单元,生成门控循环神经网络隐藏变量 e_t ;

[0024] 将所述门控循环神经网络隐藏变量 e_t 与 $t-1$ 时刻的随机隐变量 z_{t-1} 进行拼接;

[0025] 将拼接后的 e_t 与 z_{t-1} 输入变分网络的全连接层、软加激活函数层和线性层获得 t 时刻的随机隐变量 z_t 的均值和标准差;

[0026] 基于所述随机隐变量 z_t 的均值和标准差,采用平面标准化流拟合得到随机隐变量 z_t 。

[0027] 进一步地,在本申请的一个实施例中,所述生成网络通过随机隐变量连接技术连接多时间序列片段的随机隐变量获得每一时间序列片段数据的重构值,包括:

[0028] 采用线性高斯状态空间模型连接相邻时刻的随机隐变量;

[0029] 将 t 时刻的随机隐变量 z_t 和 $t-1$ 时刻的门控循环神经网络隐藏变量 d_{t-1} 输入生成网络的门控循环神经网络单元生成门控循环神经网络隐藏变量 d_t ;

[0030] 将所述门控循环神经网络隐藏变量 d_t 输入生成网络的全连接层、软加激活函数层和线性层生成重构值的均值和标准差;

[0031] 通过所述重构值的均值和标准差计算输入数据的重构值。

[0032] 进一步地,在本申请的一个实施例中,还包括:

[0033] 采用最大化ELBO训练所述变分网络和所述生成网络并获取多时间序列片段数据的重构误差;

[0034] 通过所述多时间序列片段数据的重构误差计算多时间序列片段数据的每一时间

序列片段数据的重构概率；

[0035] 基于多时间序列片段数据的每一时间序列片段数据的重构概率，采用POT (Peaks-Over-Threshold, 阈值尖峰模型) 方法确定所述异常阈值。

[0036] 进一步地，在本申请的一个实施例中，所述对所述异常结果进行分析，包括：

[0037] 将所述异常时刻对应的时间序列片段数据的重构概率分解到单时间序列；

[0038] 对每个单时间序列的重构概率进行升序排列。

[0039] 为达到上述目的，本申请第二方面实施例提出了一种多时间序列数据异常检测装置，包括：

[0040] 获取分割模块，用于获取待处理数据，并将所述待处理数据分割成多时间序列片段数据；

[0041] 第一计算模块，用于通过离线训练模型计算所述多时间序列片段数据的每一时间序列片段数据的重构值；

[0042] 第二计算模块，基于所述每一时间序列片段数据的重构值计算每一时间序列片段数据的重构概率；

[0043] 比较模块，用于在线计算异常时刻数据的重构概率并与所述异常阈值比较获取异常结果；

[0044] 分析模块，对所述异常结果进行分析。

[0045] 本申请实施例的多时间序列数据异常检测装置，获取待处理数据，并将待处理数据分割成多时间序列片段数据；通过离线训练模型计算多时间序列片段数据的每一时间序列片段数据的重构值；基于每一时间序列片段数据的重构值计算每一时间序列片段数据的重构概率；根据异常时刻对应的时间序列片段数据的重构概率与异常阈值比较获取异常结果，并对异常结果进行分析。由此，在考虑多时间序列随机性和时间依赖性的前提下，对多时间序列的历史正常模式进行学习，从而实现了对多时间序列数据更加准确的多时间序列数据异常检测，并使模型的输出结果具有可解释性。

[0046] 本申请的另一方面在于，提供一种电子设备，包括：

[0047] 至少一个处理器；以及

[0048] 与所述至少一个处理器耦合连接的存储器；其中，

[0049] 所述存储器存储有计算机程序，所述计算机程序能够被所述至少一个处理器执行，以实现本申请所述的方法。

[0050] 本申请的另一方面在于，提供一种计算机可读存储介质，所述计算机可读存储介质用于存储计算机程序，当所述计算机程序被执行时，能够实现本申请所述的方法。

[0051] 本申请附加的方面和优点将在下面的描述中部分给出，部分将从下面的描述中变得明显，或通过本申请的实践了解到。

附图说明

[0052] 本申请上述的和/或附加的方面和优点从下面结合附图对实施例的描述中将变得明显和容易理解，其中：

[0053] 图1为本申请实施例的一种多时间序列数据异常检测方法的流程示例图；

[0054] 图2是本申请实施例构建网络模型中变分网络模型示意图；

- [0055] 图3是本申请实施例构建网络模型中生成网络模型示意图；
 [0056] 图4是本申请实施例构建网络模型中变分网络结构图；
 [0057] 图5是本申请实施例构建网络模型中生成网络结构图；
 [0058] 图6为本申请实施例提供的一种多时间序列数据异常检测装置的结构示意图。

具体实施方式

[0059] 下面详细描述本申请的实施例，所述实施例的示例在附图中示出，其中自始至终相同或类似的标号表示相同或类似的元件或具有相同或类似功能的元件。下面通过参考附图描述的实施例是示例性的，旨在用于解释本申请，而不能理解为对本申请的限制。

[0060] 下面参照附图描述根据本申请实施例提出的多时间序列数据异常检测方法装置。

[0061] 具体地，表1列举了现有的无监督算法及其不足，LSTM-NDT算法采用确定性方法对多时间序列进行建模，DAGMM采用随机方法但忽略了时间序列数据的时间依赖性，LSTM-VAE只是简单的将长短期记忆网络和随机变量组合在一起，这些方法都不能很准确地解决多时间序列的随机性和时间依赖性。

[0062] 表1现有无监督算法及其不足

工作	算法	检测对象	维度	不足
[0063] LSTM-NDT	长短期记忆网络	卫星	25~55	无法抓取时间序列的随机性
DAGMM	自编码器+高斯混合模型	心率疾病	279	忽略了时间序列的时间依赖性
LSTM-VAE	变分自编码器+长短期记忆网络	机器人辅助系统	17	忽略了模型内随机变量间的时间依赖性

[0064] 相关技术在实际应用过程中效果并不好，无法达到满意的检测准确度。首先，有监督的方法需要大量的标注数据进行模型训练并且只能检测出已标注的异常类型，因此，在实际的运维工作中，多时间序列异常检测必须是无监督的方式。其次，卫星、机器等具有不同的软件逻辑控制，并以非常复杂的方式与周围环境、操作员和其他系统进行交互。这些复杂的行为使得多时间序列表现出很大的随机性和强烈的时间依赖性。

[0065] 目前现有的无监督的多时间序列异常检测方法具有如下局限性：(1) 采用确定性方法对多时间序列进行建模；(2) 采用随机方法但忽略了时间序列数据的时间依赖性；(3) 简单的将长短期记忆网络和随机变量组合在一起；这些方法都不能很准确地解决多时间序列的随机性和时间依赖性。最后，这些研究都只对多时间序列进行异常判断，无法对检测的异常结果提供进一步的解释。

[0066] 针对上述问题，本申请提出一种多时间序列数据异常检测方法，获取待处理数据，并将待处理数据分割成多时间序列片段数据；通过离线训练模型计算多时间序列片段数据的每一时间序列片段数据的重构值；基于每一时间序列片段数据的重构值计算每一时间序列片段数据的重构概率；根据异常时刻对应的时间序列片段数据的重构概率与异常阈值比较获取异常结果，并对异常结果进行分析。由此，在考虑多时间序列随机性和时间依赖性的

前提下,对多时间序列的历史正常模式进行学习,从而实现对多时间序列数据更加准确的多时间序列数据异常检测,并使模型的输出结果具有可解释性。

[0067] 图1为本申请实施例的一种抗散射成像多时间序列数据异常检测方法的流程示例图。如图1所示,该抗散射成像多时间序列数据异常检测方法,包括:

[0068] 步骤101,获取待处理数据,并将待处理数据分割成多时间序列片段数据。

[0069] 在本申请实施例中,获取待处理数据,可以理解的是待处理数据为待检测数据比如一台机器有CPU使用率、网络使用率、内存使用率、磁盘使用率等。

[0070] 在本申请实施例中,对待处理数据进行数据清洗,并对清洗后的数据进行基于均值方差的标准化处理,进一步提高后续检测的准确性。

[0071] 具体地,在生产环境中,数据监控平台可能未能成功采样一些时间点的历史监控数据,或者错误地将一些数据存储为非数值型数据——这些数据被分别称为缺失数据和脏数据。由于这些数据会降低模型的训练和检测的性能,因此需要对数据进行预处理。首先,本实施例将这些缺失数据和脏数据统一设置为零。其次,为了消除不同指标的振幅影响,本申请对数据进行基于均值方差的标准化处理。

[0072] 在本申请实施例中,通过滑动窗口算法将待处理数据分割为长度为 $t+1$ 的多时间序列片段数据, t 为正整数。

[0073] 具体地,对输入数据进行滑动窗口处理,每个窗口是长度为 $t+1$ 的时间序列片段:

[0074] $x_{t-T:t} (\in \mathbb{R}^{M \times (T+1)}) = \{x_{t-T}, x_{t-T+1}, \dots, x_t\}$

[0075] 本申请实施例中,多时间序列数据可以用 x 表示, $x = \{x_1, x_2, \dots, x_N\}$, N 是数据 x 的长度,任一时刻 t 的数据点 $x_t \in \mathbb{R}^M$ 是一个 M 维的实数(\mathbb{R})向量: $x_t = [x_t^1, x_t^2, \dots, x_t^M]$,所以 $x \in \mathbb{R}^{M \times N}$ 。多时间序列异常检测工作的目的是判断 x_t 异常与否。

[0076] 步骤102,通过离线训练模型计算多时间序列片段数据的每一时间序列片段数据的重构值。

[0077] 在本申请的实施例中,离线训练模型,包括变分网络和生成网络。

[0078] 在本申请的实施例中,变分网络将多时间序列片段数据的每一时间序列片段数据映射到随机隐变量;生成网络通过随机隐变量连接技术连接多时间序列片段的随机隐变量获得每一时间序列片段数据的重构值。

[0079] 在本申请的实施例中,将多时间序列片段数据输入变分网络进行处理,获取多时间序列片段的随机隐变量,包括:将 t 时刻的输入数据 x_t 和 $t-1$ 时刻的门控循环神经网络隐藏变量 e_{t-1} 输入变分网络的门控循环神经网络单元,生成门控循环神经网络隐藏变量 e_t ;将所述门控循环神经网络隐藏变量 e_t 与 $t-1$ 时刻的随机隐变量 z_{t-1} 进行拼接;将拼接后的 e_t 与 z_{t-1} 输入变分网络的全连接层、软加激活函数层和线性层获得 t 时刻的随机隐变量 z_t 的均值和标准差;基于随机隐变量 z_t 的均值和标准差,采用平面标准化流拟合得到随机隐变量 z_t 。

[0080] 在本申请的实施例中,平面标准化流拟合具体为:将初始的随机隐变量 z_{t0} 经过平面映射转化拟合得到最终的随机隐变量 z_t 。

[0081] 如图2所示的变分网络模型示意图, $t-T$ 时刻的训练数据 x_{t-T} 作为输入数据,生成 e_{t-T} ,通过 e_{t-T} 生成 $t-T$ 时刻的随机隐变量 z_{t-T} ,因此, $t-T:t$ 时间片段内的其余随机隐变量 $z_{t-T:t}$ 均采用上述方法不断迭代,最终得到 z_t 。

[0082] 在本申请的实施例中,在本申请的实施例中,将多时间序列片段的随机隐变量输入生成网络进行连接处理,获取输入数据的重构值,包括:采用线性高斯状态空间模型连接相邻时刻的随机隐变量;将 t 时刻的随机隐变量 z_t 和 $t-1$ 时刻的门控循环神经网络隐藏变量 d_{t-1} 输入生成网络的门控循环神经网络单元生成门控循环神经网络隐藏变量 d_t ;将门控循环神经网络隐藏变量 d_t 输入生成网络的全连接层、软加激活函数层和线性层生成重构值的均值和标准差;通过重构值的均值和标准差计算输入数据的重构值。

[0083] 如图3所示的生成网络模型示意图,在变分网络得到的 $t-T$ 时刻的 $t-T$ 时刻的随机隐变量作为输入数据,生成 d_{t-T} ,通过 d_{t-T} 生成 $t-T$ 时刻的 x_{t-T} 的重构值 x'_{t-T} ,因此, $t-T:t$ 时间片段内的其余重构值 $X'_{t-T:t}$ 均采用上述方法不断迭代,最终得到 x'_t 。

[0084] 具体地,使用GRU(Gated Recurrent Unit,门控循环单元)捕捉原始数据空间(X 空间)的多指标数据间的复杂时间依赖关系。其次,应用一种常用的算法——VAE(Variational Auto-Encoder,变分自编码器)将输入变量(即 X 空间中的数据)映射到随机隐(即 Z 空间)变量。再次,为了明确地建立 Z 空间中随机变量之间的时间依赖关系,本申请提出一种随机隐变量连接技术:使用Linear SSM(Linear Gaussian State Space Model,线性高斯状态空间模型)连接随机隐变量,通过将潜在变量构成马尔科夫链,从而使得变量之间具有时序关系,并将随机隐变量与GRU隐藏变量进行拼接。最后,为了帮助变分网络中的随机变量捕获输入数据的复杂分布,本实施例采用planar NF(planar Normalizing Flow,平面标准化流,planar NF通过一系列可优化的线性函数将简单分布映射为更复杂的分布)来学习 Z 空间中的非高斯后验分布。

[0085] 更具体地,如图2和3所示,本实施例的模型网络由一个变分网络和一个生成网络组成,本申请提供的变分网络和生成网络为随机循环神经网络,创新性地使用随机隐变量连接技术来挖掘模型的潜在表示,在海量历史数据中无监督地学习多时间序列的正常模式,充分考虑了多时间序列的时序依赖和随机性。。

[0086] 其中,变分网络的目的是为了优化逼近后验分布,获得良好的随机隐变量。如图4所示的变分网络结构图,在时间 t ,原始输入数据点 x_t 和 $t-1$ 时刻的GRU隐藏变量 e_{t-1} 被输入到GRU单元,以生成确定性的GRU隐藏变量 e_t 。 e_t 的目的是捕获 x_t 及其之前的 X 空间中数据的长期复杂时间依赖关系。然后, e_t 与 $t-1$ 时刻的随机隐变量 z_{t-1} 拼接,先进入Dense层(全连接层 h^0),然后进入Soft plus层(软加激活函数层)和线性层,生成随机变量 z_t 的均值 μ_{z_t} 和标准差 σ_{z_t} 。因此, Z 空间的变量是明确依赖的。为了学习到复杂的非高斯概率分布 $q_\phi(z_t|x_t)$,本方法使用planar NF去拟合 z_t ,即把一个初始的 Z 变量 z_t^0 经过一系列的平面映射转化 f^k 得到最终的 z_t^k (即 z_t)。

[0087] 在生成网络中,使用隐藏变量 $z_{t-T:t}$ 来重构输入数据 $x_{t-T:t}$,精确的随机隐变量可以最大限度地减少重构损失。具体的,如图5所示的生成网络结构图,在生成网络中,首先使用线性高斯SSM(Linear Gaussian State Space Model)连接 Z 变量,使其存在明显依赖关系(T_0, O_0 为SSM参数,可以在模型训练过程中自动学习)。在时间 t , z_t 和确定性GRU变量 d_{t-1} 一起被输入到GRU单元,生成 d_t 。 d_t 进入Dense层(全连接层, h^0 为网络参数,可以在模型训练过程中自动学习)和Soft plus层(软加激活函数层),生成变量 x_t' 的均值 $\mu_{x_t'}$ 和标准差 $\sigma_{x_t'}$ 。 x_t' 的重构值 x_t' 是从高斯分布 $N(\mu_{x_t'}, \sigma_{x_t'}^2 I)$ 中采样得到。如果在时间 t 发生异常,重构值 x_t' 和原始值

x_t 的差距会变得很大,因此可以通过二者的差异值检测异常。

[0088] 步骤103,基于每一时间序列片段数据的重构值计算每一时间序列片段数据的重构概率。

[0089] 步骤104,根据异常时刻对应的时间序列片段数据的重构概率与异常阈值比较获取异常结果,并对异常结果进行分析。

[0090] 在本申请的实施例中,采用最大化ELBO训练变分网络和所述生成网络并获取多时间序列片段数据的重构误差;通过多时间序列片段数据的重构误差计算多时间序列片段数据的每一时间序列片段数据的重构概率;基于多时间序列片段数据的每一时间序列片段数据的重构概率,采用POT方法确定异常阈值。

[0091] 在本申请的实施例中,将异常时刻对应的时间序列片段数据的重构概率分解到单时间序列;对每个单时间序列的重构概率进行升序排列。

[0092] 本申请的多时间序列数据异常检测方法还包括:利用增量的训练数据,采用最大化ELBO训练变分网络和生成网络对网络模型进行优化,使得训练的模型可以适应最新的异常模式。

[0093] 模型的生成网络和变分网络会被同时进行训练以优化模型中的参数。本实施例通过最大化ELBO (evidence of lower bound, 证据下限) 来训练模型,模型的训练目标函数为:

$$[0094] \quad \tilde{L}(x_{t-T:t}) \approx \frac{1}{L} \sum_{l=1}^L [\log p_{\theta}(x'_{t-T:t} | z_{t-T:t}^{(l)}) + \log p_{\theta}(z_{t-T:t}^{(l)}) - \log q_{\varphi}(z_{t-T:t}^{(l)} | x_{t-T:t})]$$

[0095] 其中L是采样长度。公式第1项是负重构误差:

$$[0096] \quad \log p_{\theta}(x'_{t-T:t} | z_{t-T:t}^{(l)}) = \sum_{i=t-T}^t \log p_{\theta}(x'_i | z_{t-T:i}) \quad , X_i \text{ 的后验概率为:}$$

$p_{\theta}(x'_i | z_{t-T:t}) \sim N(\mu_{x'_i}, \sigma_{x'_i}^2 I)$, 第2项与第3项的和是Kullback-Leibler divergence (KL散度)。

其中,第2项 $\log p_{\theta}(z_{t-T:t}) = \sum_{i=t-T}^t \log p_{\theta}(z_i | z_{i-1})$, z_i 通过线性高斯SSM与标准多元正态分布初始化得到。第3项是近似变分网络中 z_i 的真实后验分布:

$$[0097] \quad -\log q_{\varphi}(z_{t-T:t}^{(l)} | x_{t-T:t}) = -\sum_{i=t-T}^t \log q_{\varphi}(z_i | z_{i-1}, x_{t-T:t}) \quad , z_i \text{ (即 } z_i^K \text{) 通过planar NF得到: } z_i^K = f^K(f^{K-1}(\dots f^1(z_i^0))) \text{, } z_i^0 = \mu_{z_i} + \xi_i \sigma_{z_i} \text{, } \xi_i \sim N(0, I)$$

[0098] 在更优选的实施例中,为进一步确认某个时间点的数据 (x_t) 是否异常,采用重构概率 $\log p_{\theta}(x'_t | z_{t-T:t})$ 来表示异常分数 S_t , $\log p_{\theta}(x'_t | z_{t-T:t})$ 由重构误差 $\log p_{\theta}(x'_{t-T:t} | z_{t-T:t}^{(l)}) = \sum_{i=t-T}^t \log p_{\theta}(x'_i | z_{t-T:i})$ 得到。高分数表示输入数据 x_t 可以很好地被重构。如果输入数据遵循指标的正常模式,该数据就可以被很好地重构。相反的,分数越小表示重构成功的可能性越小,因此数据为异常的可能性越大。所以,如果 S_t 低于异常阈值,那么 x_t 被标记为异常;否则 x_t 是正常的。

[0099] 在模型离线训练过程中,模型会计算每一个训练数据的重构概率,所有的重构概率形成一个新的指标序列: $\{S_1, S_2, \dots, S_{N'}\}$ (N' 为训练数据长度)。本申请使用POT设置异常阈值 thF 。POT的基本思想是用带参数的广义帕累托分布(GPD)来计算数据尾部的概率分布。与专注于数据“高值部分”的经典POT应用不同,本申请的重构概率位于低值部分。因此,调

整后的GPD函数如下：

$$[0100] \quad \bar{F}(s) = P(\text{th} - S > s | S < \text{th}) \sim (1 + \frac{\gamma s}{\beta})^{-\frac{1}{\gamma}}$$

[0101] th 是重构概率的初始阈值,通过一个低比例数值确定。 γ 和 β 是GPD的形状和比例参数, S 是 $\{S_1, S_2, \dots, S_{N'}\}$ 中的任意值。小于阈值 th 的值被表示为 $\text{th}-S$,最终的阈值为 thF ：

$$[0102] \quad \text{thF} \cong \text{th} - \frac{\hat{\beta}}{\hat{\gamma}} \left(\left(\frac{qN'}{N'_{\text{th}}} \right)^{-\hat{\gamma}} - 1 \right)$$

[0103] $\hat{\gamma}$ 和 $\hat{\beta}$ 可以通过最大似然估计得到的 γ 和 β 的估计值, q 是 $S < \text{th}$ 的期望概率, N' 是输入数据长度, N'_{th} 是 $S_i < \text{th}$ 的长度。对于POT方法,只需要调整低比例数值 N'_{th}/N' 和 q 两个参数:例如低比例数值小于7%, q 取值 10^{-4} 。

[0104] 对异常结果提供解释,异常解释的目的是分析多时间序列数据的每个单时间序列的重构概率,并用排序靠前几个单时间序列来解释检测的实体异常。虽然模型计算的是 x_t 的整体重构概率,但根据 $p_{\theta}(x'_i | z_{t-T:t}) \sim N(\mu_{x'_i}, \sigma_{x'_i}^2 I)$,可以得出

$p_{\theta}(x'_i | z_{t-T:t}) = \prod_{i=1}^M p_{\theta}(x_t^i | z_{t-T:t})$,因此 x_t 的重构概率可以被分解为:

$$[0105] \quad \log(p_{\theta}(x'_i | z_{t-T:t})) = \sum_{i=1}^M \log(p_{\theta}(x_t^i | z_{t-T:t}))$$

[0106] $S_t = \sum_{i=1}^M S_t^i$, $S_t^i = \log(p_{\theta}(x_t^i | z_{t-T:t}))$, S_t^i 是 x_t^i 的重构概率。对于异常输入数据 x_t 可以通过评估 x_t 每个每个单时间序列的重构概率,即异常贡献来解释它。。对 S_t^i 进行升序排序,形成列表 AS_t 。在 AS_t 中,排名越高, S_t^i 越小, x_t^i 对 x_t 的贡献越大。排序结果将作为异常解释提供给运维人员,排序最靠前的单时间序列可以为运维人员理解和分析检测到的实体异常提供足够的线索。

[0107] 本实施例在2个公开数据集(2个数据集名字:SMAP和MSL,数据公开链接:<https://github.com/khundman/telemanom>)和1个互联网公司(数据集名字:SMD,数据公开链接:<https://github.com/NetManAIops/OmniAnomaly/tree/master/ServerMachineDataset>)实际生产环境中测试。本实施例可以准确地检测并解释异常,达到0.86的检测F分数(F-score)和0.89的异常解释准确度。如表格2所示的实验结果证明,提出的方案比学术界现有的最先进算法都要好。

[0108] 表2现有算法和本实施例算法的结果对比

[0109]

方法	SMAP			MSL		
	准确度 (Precision)	召回率 (Recall)	F分数 (F-score)	准确度 (Precision)	召回率 (Recall)	F分数 (F-score)
LSTM-NDT	0.8965	0.8846	0.8905	0.5934	0.5374	0.5640
DAGMM	0.5845	0.9058	0.7105	0.5412	0.9934	0.7007
LSTM-VAE	0.8551	0.6366	0.7298	0.5257	0.9546	0.6780
实施例2	0.7416	0.9776	0.8434	0.8867	0.9117	0.8989
方法	SMD			结果汇总		
	准确度 (Precision)	召回率 (Recall)	F分数 (F-score)	准确度 (Precision)	召回率 (Recall)	F分数 (F-score)
LSTM-NDT	0.5684	0.6438	0.6037	0.7598	0.7794	0.7694
DAGMM	0.5951	0.8782	0.7094	0.5835	0.9042	0.7093
LSTM-VAE	0.7922	0.7075	0.7842	0.7782	0.7075	0.7411
实施例2	0.8334	0.9449	0.8857	0.7797	0.9586	0.8599

[0110] 本申请实施例的多时间序列数据异常检测方法,获取待处理数据,并将待处理数据分割成多时间序列片段数据;通过离线训练模型计算多时间序列片段数据的每一时间序列片段数据的重构值;基于每一时间序列片段数据的重构值计算每一时间序列片段数据的重构概率;根据异常时刻对应的时间序列片段数据的重构概率与异常阈值比较获取异常结果,并对异常结果进行分析。由此,在考虑多时间序列随机性和时间依赖性的前提下,对多时间序列的历史正常模式进行学习,从而实现对多时间序列数据更加准确的多时间序列数据异常检测,并使模型的输出结果具有可解释性。

[0111] 为了实现上述实施例,本申请还提出一种多时间序列数据异常检测装置。

[0112] 图6为本申请实施例提供的一种多时间序列数据异常检测装置的结构示意图。

[0113] 如图6所示,该装置包括:获取分割模块601、第一计算模块602、第二计算模块603、比较模块604和分析模块605。

[0114] 获取分割模块601,用于获取待处理数据,并将所述待处理数据分割成多时间序列片段数据。

[0115] 第一计算模块602,用于通过离线训练模型计算所述多时间序列片段数据的每一时间序列片段数据的重构值。

[0116] 第二计算模块603,基于所述每一时间序列片段数据的重构值计算每一时间序列片段数据的重构概率。

[0117] 比较模块604,用于在线计算异常时刻数据的重构概率并与异常阈值比较获取异常结果。

[0118] 分析模块605,对异常结果进行分析。

[0119] 需要说明的是,前述对方法实施例的解释说明也适用于该实施例的装置,此处不再赘述。

[0120] 本申请实施例的多时间序列数据异常检测装置,获取待处理数据,并将待处理数据分割成多时间序列片段数据;通过离线训练模型计算多时间序列片段数据的每一时间序列片段数据的重构值;基于每一时间序列片段数据的重构值计算每一时间序列片段数据的重构概率;根据异常时刻对应的时间序列片段数据的重构概率与异常阈值比较获取异常结果,并对异常结果进行分析。由此,在考虑多时间序列随机性和时间依赖性的前提下,对多时间序列的历史正常模式进行学习,从而实现对多时间序列数据更加准确的多时间序列数

据异常检测,并使模型的输出结果具有可解释性。

[0121] 在本说明书的描述中,参考术语“一个实施例”、“一些实施例”、“示例”、“具体示例”或“一些示例”等的描述意指结合该实施例或示例描述的具体特征、结构、材料或者特点包含于本申请的至少一个实施例或示例中。在本说明书中,对上述术语的示意性表述不必针对的是相同的实施例或示例。而且,描述的具体特征、结构、材料或者特点可以在一个或多个实施例或示例中以合适的方式结合。此外,在不相互矛盾的情况下,本领域的技术人员可以将本说明书中描述的不同实施例或示例以及不同实施例或示例的特征进行结合和组合。

[0122] 此外,术语“第一”、“第二”仅用于描述目的,而不能理解为指示或暗示相对重要性或者隐含指明所指示的技术特征的数量。由此,限定有“第一”、“第二”的特征可以明示或者隐含地包括至少一个该特征。在本申请的描述中,“多个”的含义是至少两个,例如两个,三个等,除非另有明确具体的限定。

[0123] 流程图中或在此以其他方式描述的任何过程或方法描述可以被理解为,表示包括一个或更多个用于实现定制逻辑功能或过程的步骤的可执行指令的代码的模块、片段或部分,并且本申请的优选实施方式的范围包括另外的实现,其中可以不按所示出或讨论的顺序,包括根据所涉及的功能按基本同时的方式或按相反的顺序,来执行功能,这应被本申请的实施例所属技术领域的技术人员所理解。

[0124] 在流程图中表示或在此以其他方式描述的逻辑和/或步骤,例如,可以被认为是用于实现逻辑功能的可执行指令的定序列列表,可以具体实现在任何计算机可读介质中,以供指令执行系统、装置或设备(如基于计算机的系统、包括处理器的系统或其他可以从指令执行系统、装置或设备取指令并执行指令的系统)使用,或结合这些指令执行系统、装置或设备而使用。就本说明书而言,“计算机可读介质”可以是任何可以包含、存储、通信、传播或传输程序以供指令执行系统、装置或设备或结合这些指令执行系统、装置或设备而使用的装置。计算机可读介质的更具体的示例(非穷尽性列表)包括以下:具有一个或多个布线的电连接部(电子装置),便携式计算机盘盒(磁装置),随机存取存储器(RAM),只读存储器(ROM),可擦除可编程只读存储器(EEPROM或闪速存储器),光纤装置,以及便携式光盘只读存储器(CDROM)。另外,计算机可读介质甚至可以是可在其上打印所述程序的纸或其他合适的介质,因为可以例如通过对纸或其他介质进行光学扫描,接着进行编辑、解译或必要时以其他合适方式进行处理来以电子方式获得所述程序,然后将其存储在计算机存储器中。

[0125] 应当理解,本申请的各部分可以用硬件、软件、固件或它们的组合来实现。在上述实施方式中,多个步骤或方法可以用存储在存储器中且由合适的指令执行系统执行的软件或固件来实现。如,如果用硬件来实现和在另一实施方式中一样,可用本领域公知的下列技术中的任一项或他们的组合来实现:具有用于对数据信号实现逻辑功能的逻辑门电路的离散逻辑电路,具有合适的组合逻辑门电路的专用集成电路,可编程门阵列(PGA),现场可编程门阵列(FPGA)等。

[0126] 本技术领域的普通技术人员可以理解实现上述实施例方法携带的全部或部分步骤是可以通程序来指令相关的硬件完成,所述的程序可以存储于一种计算机可读存储介质中,该程序在执行时,包括方法实施例的步骤之一或其组合。

[0127] 此外,在本申请各个实施例中的各功能单元可以集成在一个处理模块中,也可以

是各个单元单独物理存在,也可以两个或两个以上单元集成在一个模块中。上述集成的模块既可以采用硬件的形式实现,也可以采用软件功能模块的形式实现。所述集成的模块如果以软件功能模块的形式实现并作为独立的产品销售或使用,也可以存储在一个计算机可读取存储介质中。

[0128] 上述提到的存储介质可以是只读存储器,磁盘或光盘等。尽管上面已经示出和描述了本申请的实施例,可以理解的是,上述实施例是示例性的,不能理解为对本申请的限制,本领域的普通技术人员在本申请的范围内可以对上述实施例进行变化、修改、替换和变型。

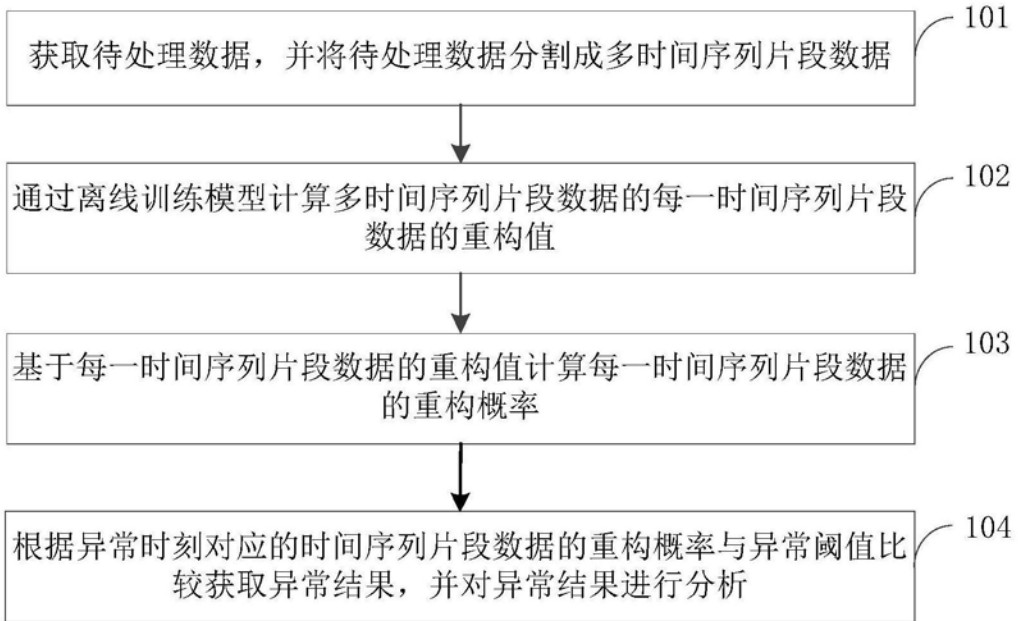


图1

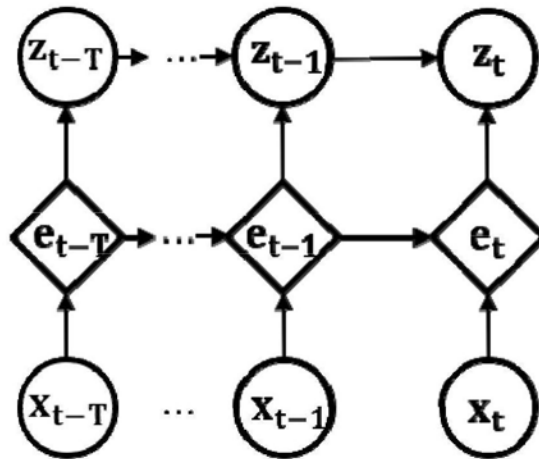


图2

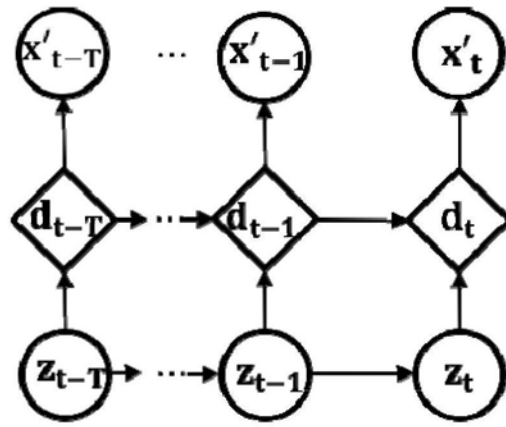


图3

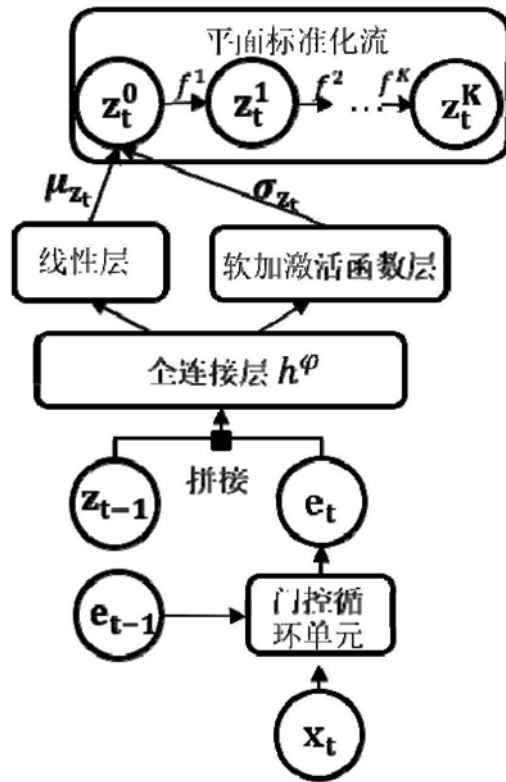


图4

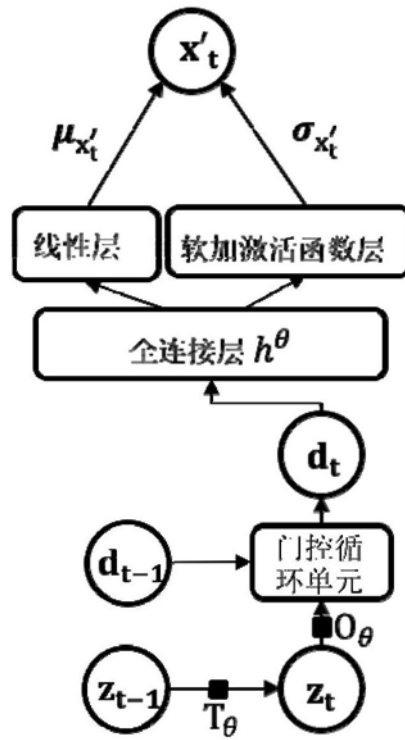


图5

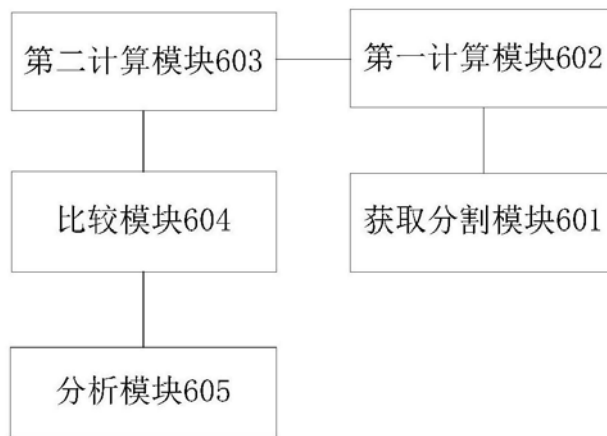


图6