基于多特征融合的应用系统监控指标异常检测方法

曹钰聪 张 俊
(大连海事大学信息科学技术学院 辽宁 大连 116026)

摘 要 为解决现有监控指标异常检测技术存在的特征学习不充分、阈值固定等问题,提出一种基于多特征融合的应用系统监控指标异常检测方法。使用 1D-CNN(1D-Convolutional Neural Network)与 SRNN(Stochastic Recurrent Neural Network)提取数据特征,引入 SE 块(Squeeze-and-Excitation)突出指标关键特征以优化特征提取,加强分类效果。以 VAE(Variational Auto-Encoder)为框架计算数据重构概率,并通过优化的极值模型计算最优异常阈值以判断异常。实验结果表明,所提方法在基于两个公开数据集的异常检测任务的 F1 评分最优达到 92%,优于目前先进的异常检测方法。

关键词 监控指标 异常检测 特征提取 变分自编码器 极值理论

中图分类号 TP391 **文献标志码** A DOI:10.3969/j.issn.1000-386x.2024.08.011

ANOMALY DETECTION METHOD FOR KPIS IN APPLICATION SYSTEMS BASED ON MULTI-FEATURE FUSION

Cao Yucong Zhang Jun
(*Information Science and Technology College*, *Dalian Maritime University*, *Dalian 116026*, *Liaoning*, *China*)

Abstract In order to solve the problems of existing KPIs anomaly detection methods, such as insufficient feature learning and fixed thresholds, we propose a anomaly detection method for KPIs in application systems based on multi-feature fusion. We used the 1D-convolutional neural network (1D-CNN) and stochastic recurrent neural network (SRNN) to extract data features, and introduced the squeeze-and-excitation (SE) block to highlight the key features of KPIs to optimize feature extraction and strengthen the classification effect. We used the variational auto-encoder (VAE) as the framework to calculate the reconstruction probability of data, and calculated the best anomaly threshold through the extreme value model to determine anomalies. Experimental results show that the proposed method can effectively detect outlier on two public datasets, with best F1 score of 92%, and has better performance than some advanced anomaly detection methods.

Keywords KPIs Anomaly detection Feature extraction Variational auto-encoder Extreme value theory

0 引 言

在信息化迅速发展的背景下,随着用户需求的日益复杂,应用系统提供的服务向着多样化发展,致使应用软件及其运行环境日趋复杂,而应用系统在运行过程中不可避免地会发生故障、入侵等异常,这些异常轻则影响用户体验,严重的则会造成不可估量的经济损失。因此,应用系统在运行过程中,准确识别异常,对系统的服务质量管理具有十分重要的意义。

应用系统的监控指标数据,如 CPU 使用率、内存使用率、磁盘使用率、网络吞吐率和网页响应时间等,能够很好地反映应用系统运行状态,也是准确检测系统异常并及时进行故障排除的关键[1]。

目前在学术界和工业界针对监控指标数据异常检测的研究和方法有很多,主要分为基于传统统计学、有监督集成式方法和无监督学习方法。

基于传统统计学方法的异常检测主要是假定用一

收稿日期:2021-04-05。国家自然科学基金项目(61976032)。**曹钰聪**,硕士生,主研领域:异常检测。**张俊**,教授。

个参数模型来学习数据的分布,使用统计学知识对数据进行预测或基于偏差计算,从而实现异常检测。如小波变换[2]、ARIMA[3]、S-ESD 和 S-H-ESD[4]方法。但在实践中,该类方法主要针对单一指标,并且在许多情况下数据分布是未知的,需要运维专家根据统计理论和运维经验,对特定监控指标数据类型的方法及参数进行调整,所以很难得到大规模的使用。

有监督集成式方法的主要思想是运用多种分类方法作为基分类器,将基分类器的分类结果作某种运算,得到最终分类结果。如孤立森林[5]和 Opprentice[6]。但是基于监督学习的方法依赖大量标签,耗费人力和时间成本,且对于数据特征的提取也依赖于经验,无法对数据特征进行充分提取。

基于无监督学习方法可以分为基于邻近度的方法如一类向量机[7](One-Class SVM)、基于聚类的方法如 DBSCAN[8],和基于生成模型的方法如变分自编码器[9](VAE)、生成对抗网络[10](Generative Adversarial Networks,GAN)。无监督学习的优点在于无须使用异常标签并且基本无须对数据分布进行前提假设。但大部分基于无监督的异常检测方法还存在以下几方面问题:(1) 监控指标是时间序列数据,存在时间依赖性,而基于聚类、密度估计的方法未考虑到这一特性,如文献[11]提出的核心为密度估计与深度自动编码混合模型(DAGMM),它通过将深度自动编码器和密度估计过程结合在一起进行端到端训练来建模多维数据的密度估计,虽能够检测异常,但忽略了时间序列固有的时间依赖性,会产生误判。(2) 实际运维中,运维人员更关注的是系统整体运行状态的监控,且复杂系统组件及其内部间一般都存在非线性、强关联耦合的相互关系[12],异常事件会传播性地反映在多个监控指标上,具有异常传播的特点。监控指标的特征重要程度会随异常事件类型的变化而变化,数据特征提取的完整程度直接影响异常检测效果[12]。如文献[13]提出的 OmniAnomaly 方法,运用随机变量连接和标准化流将 VAE 与 GRU 黏合在一起,在考虑时间依赖性和随机性的同时通过重构误差检测异常,但并没有考虑指标相关性和特征重要程度,异常检测效果受到特征提取程度的影响。(3) 监控指标数据形态各异,为监控指标设定固定阈值会导致不能有效监测未知异常,如文献[14]使用生成对抗网络同时训练编码器,并通过使用平滑注意力机制捕捉时序特征,但未能实现阈值自动调整,造成误判漏判,影响异常判断准确度。

深度学习是近年来的研究热点,其通过神经网络自动学习数据特征并优化模型,由于灵活性、鲁棒性强等特点,可将其应用在运维领域,解决上文中存在的问题。本文运用神经网络知识,在已有研究的基础上提出一种应用系统异常检测方法——CS-VAE,从系统层面对系统整体运行状态进行异常检测。具体来说:首先,CS-VAE 将 1D-CNN、SE 与 GRU 作为特征提取网络结构,捕获指标数据的相关性特征与时序特征,实现多种特征的融合,全面刻画数据特征;随后,以 VAE 模型为框架,重构应用系统监控指标数据,得到数据重构概率分数;运用极值理论 POT 方法根据重构概率分数确定最优阈值,分割异常数据子序列。本文在 SMD(Server Machine Dataset)[13]、SOD(Software Operational Data)[15]数据集上进行对比实验,实验结果表明本文方法优于最先进的基线方法,并对各组成部分进行消融实验,验证模型有效性。

1　问题定义

从图 1 可以看出在 0 ~ 1 天和 2 ~ 3 天时,系统运行正常,指标数据呈现出一定规律性,而在 1 ~ 2 天的一时间段内,由于 CPU 服务器受到资源瓶颈、故障传播影响,导致应用系统各指标发生突变,最终在应用层显示出响应时间过长的故障。针对在应用系统受到故障干扰时,各监控指标会受到故障传播的影响从而发生异于正常时的突变这一现象,本文旨在提出一个新的异常检测方法,从应用系统整体运行状态出发,对系统各组件的监控数据进行异常检测,充分考虑指标相关性、特征重要程度和时间依赖性等特点,准确检测出应用系统各监控指标的突变以检测异常。本文研究内容定义如定义 1 所示。

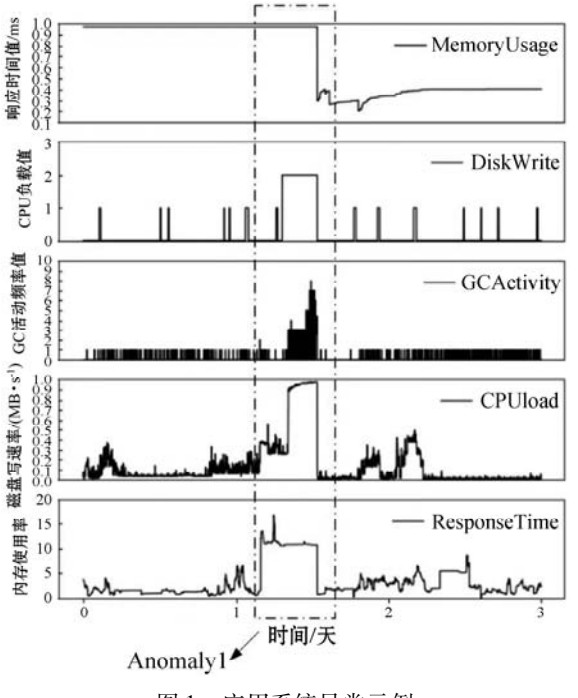


图 1　应用系统异常示例

定义 1 应用系统监控指标异常检测 对于应用系统的监控指标数据 $\boldsymbol{X}=(x_1,x_2,\cdots,x_t,x_{t+1},\cdots,x_N)^{\mathrm{T}}$，若 t 时刻指标数据 $x_t^j(0\leqslant j\leqslant M)$ 的 M 维数据突变程度总和大于异常阈值 t_{hreshold}，则 t 时刻系统处于异常状态。

2 相关工作

2.1 一维卷积神经网络

对于监控指标、传感器数据这类时间序列数据来说，一般使用一维卷积操作对其进行时间卷积，可得到序列数据的空间相关性特征。1D-CNN 与 CNN 结构相同，主要包括卷积层、池化层和全连接层，卷积核用一维卷积操作提取特征，表示为：

$$S_j^{(l+1)}(\tau)=\sigma\left(b_j^{(l)}+\sum_{f=1}^{F^{(l)}}K_{jf}^{(l)}(\tau)*S_f^{(l)}(\tau)\right) \quad (1)$$

式中：$S_j^{(l)}(\tau)$ 表示在 l 层的特征图 j；σ 表示非线性激活函数；$F^{(l)}$ 指在 l 层的特征图个数；$K_{jf}^{(l)}$ 指对第 l 层的特征图 f 进行核卷积，得到第 $l+1$ 层的特征图 f；$P^{(l)}$ 是第 l 层卷积核的长度；$b^{(l)}$ 为偏差；$*$ 表示卷积运算。卷积运算提取出数据深层特征后，通过池化层对学习到的特征图进行子采样处理，简化模型复杂度与参数，起到防止过拟合的作用。

2.2 随机门控循环单元

监控指标数据作为一种典型的时间序列数据，时间依赖性是其重要的特征。门控激活单元（Gated Recurrent Unit，GRU）[16] 是一种特殊的循环神经网络结构，通过重置门与更新门的门控机制控制输入、记忆等信息。而根据监控指标数据具有随机性这一特殊特点，Fraccaro 等[17] 提出了一种结合 GRU 和状态空间模型（State Space Models，SSM）的随机序列神经生成模型 SRNN，用于准确捕捉序列的随机性与时间依赖性。SGRU 计算过程如下：

$$\boldsymbol{r}_t=\mathrm{sigmoid}(\boldsymbol{W}_r\boldsymbol{x}_t+\boldsymbol{U}_r\boldsymbol{h}_{t-1}+\boldsymbol{b}_r) \quad (2)$$

$$\boldsymbol{u}_t=\mathrm{sigmoid}(\boldsymbol{W}_u\boldsymbol{x}_t+\boldsymbol{U}_u\boldsymbol{h}_{t-1}+\boldsymbol{b}_u) \quad (3)$$

$$\boldsymbol{h}_t=\boldsymbol{u}_t\odot\tanh(\boldsymbol{W}_h\boldsymbol{x}_t+\boldsymbol{U}_h(\boldsymbol{r}_t\odot\boldsymbol{h}_{t-1})+\boldsymbol{b}_h)+(\boldsymbol{1}-\boldsymbol{u}_t)\odot\boldsymbol{h}_{t-1} \quad (4)$$

式中：$\boldsymbol{h}_t$ 表示 t 时刻隐藏层的状态；$\boldsymbol{r}_t$ 决定哪些信息需要忽略；$\boldsymbol{W}$、$\boldsymbol{U}$ 为权重矩阵；$\boldsymbol{b}$ 为偏置；$\odot$ 表示点乘运算；$\boldsymbol{u}_t$ 决定哪些值需要更新；$\boldsymbol{h}_t$ 决定模型的输出。SSM 将 GRU 的输出 $\boldsymbol{h}_t$ 与隐空间随机变量级联，使隐变量包含 GRU 建模的时间依赖性。

2.3 变分自编码器

VAE 是一种深度生成模型，已经广泛应用于异常检测，包含编码器、解码器和损失函数三部分。编码器 $q_\phi(z\mid x)$ 学习正常状态下观测数据的分布信息，将观测数据压缩到隐变量空间中，解码器 $p_\theta(x\mid z)$ 根据隐变量状态重构出数据分布，得到最小损失函数以训练模型；将真实场景下的观测数据输入到训练好的 VAE 中，得到被 VAE 重构的正常数据分布，正常数据的重构概率相对高于异常数据的重构概率，根据重构概率的大小来判定是否异常。由于观测数据 x 的分布未知，且隐变量分布也无法直接获取，VAE 假设编码器 $q_\phi(z\mid x)$ 服从高斯分布用来代替无法确定的真实后验分布 $p_\theta(x\mid z)$，通过学习高斯分布参数，经解码器 $p_\theta(x\mid z)$ 即可得到原始数据分布，为使编码器 $q_\phi(z\mid x)$ 与真实后验分布 $p_\theta(x\mid z)$ 近似相等，采用 KL 散度衡量相似度，KL 散度损失函数表示为：

$$\min_{\phi,\theta}D_{\mathrm{KL}}(q_\phi(z\mid x)\parallel p_\theta(x\mid z)) \quad (5)$$

最小化参数 ϕ 仍需要计算 x 的分布，所以进一步引入 ELBO，将对参数 ϕ 的训练变化为：

$$\mathrm{ELBO}(\phi)=E_q[\log p_\theta(x,z)]-E_q[\log q_\phi(z\mid x)] \quad (6)$$

将 KL 散度与 ELBO 相结合得到 VAE 的损失函数，利用随机梯度对参数进行优化，损失函数表示为：

$$L_{\mathrm{vae}}(\phi,\theta;x)=-D_{\mathrm{KL}}(q_\phi(z\mid x)\parallel p_\theta(z))+E_{q_\varphi(z\mid x)}[\log p_\theta(x\mid z)] \quad (7)$$

2.4 极值理论

目前已经有许多方法用来确定异常阈值。但是监控指标数据不一定遵循常见的分布，因此对数据分布的建模存在困难。极值理论[18] 是用来对极值的分布进行建模的工具，通过推断极值分布来解决阈值确定问题，而无须对原始分布进行假设。对于随机变量 X，$F(x)=P(X\leqslant x)$ 是其累积分布函数。定义 $\tilde{F}(x)=1-F(x)=P(X>x)$ 为其分布的尾部分布。由于变量的分布未知，可以采用将 EVD（Extreme Value Distribution）分布拟合到未知的输入分布尾部，评估潜在极端事件的可能性。EVD 分布表示为：

$$G_\gamma:y\to\exp(-(1+\gamma y)^{-\frac{1}{\gamma}})\quad \gamma\in\mathbf{R},1+\gamma y>0 \quad (8)$$

式中：γ 是分布的极值指数。给定阈值 t，将超出该阈值的部分写作 $X-t$，POT（Peaks-Over-Threshold）方法基于阈值 t，将广义帕累托分布（Generalized Pareto Distribution，GPD）拟合到 $X-t$ 上，其分布函数表示为：

$$F_t(x)=P(X-t>x\mid X>t)\sim\left(1+\frac{\gamma x}{\sigma(t)}\right)^{-\frac{1}{\gamma}} \quad (9)$$

通过极大似然方法确定式（9）最优参数 $\hat{\sigma}$、$\hat{\gamma}$，最

优阈值通过式(10)得出。

$$z_q \simeq t + \frac{\hat{\sigma}}{\hat{\gamma}}\left(\left(\frac{qn}{N_t}\right)^{-\hat{\gamma}} - 1\right) \tag{10}$$

式中:t 是初始阈值;q 是期望的概率;n 为数据个数;N_t 为峰值的个数,即 $X_i > t$ 的个数。

3　CS-VAE 方法

3.1　方法概述

本文提出的应用系统监控指标异常检测方法 CS-VAE 是一种基于重构的异常检测方法,基本思想为利用正常数据训练模型,再利用训练好的模型重建存在异常的数据,得到数据的重构误差,并通过阈值或其他方法分割出异常发生的子序列。CS-VAE 主要包含四个步骤:监控指标数据预处理,重构概率计算,最优异常阈值计算,系统异常时刻判断。图 2 为 CS-VAE 方法整体结构。

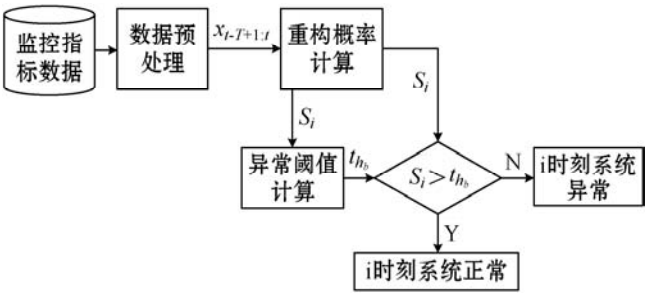


图 2　CS-VAE 方法整体结构

3.2　异常检测算法设计

异常检测算法主要由以下几种方法构成,分别是:以 VAE 为框架的重构概率计算模型,改进的 EVT 最优阈值确定方法,最后将得到的重构概率与确定的异常最优阈值相比较,来判断系统运行状态是否异常,如算法 1 所示。

算法 1　异常检测算法

输入:正常监控指标数据 $\boldsymbol{X}_{\text{normal}} \in \mathbf{R}^{(N\times M)}$,存在异常的监控指标数据 $\boldsymbol{X}_{\text{abnormal}} \in \mathbf{R}^{(N\times M)}$。

输出:异常监控指标数据集 A_D。

1.　初始化 $A_D \leftarrow \varnothing$

2.　$\phi, \theta \leftarrow$利用正常数据训练模型 $\boldsymbol{X}_{\text{normal}} \in \mathbf{R}^{(N\times M)}$

3.　调用算法 2 计算正常指标数据 $\boldsymbol{X}_{\text{normal}}$ 的重构概率分数 $S_{\text{normal}} = F_{\text{rb}}(\boldsymbol{X}_{\text{normal}})$

4.　通过正常数据重构概率计算最优阈值 $t_{h_b} = F_{\text{EVT}}(S_{\text{normal}})$

5.　for $i = 1$ to N do

6.　　计算存在异常的指标数据 $\boldsymbol{X}_{\text{abnormal}}$ 的重构概率分数 $S_i = F_{\text{rb}}(\boldsymbol{X}_{\text{abnormal}})$

7.　　if $S_i < t_{h_b}$ then

8.　　$A_D \leftarrow A_D \cup (i, x_i)$

9.　　else $i \leftarrow i+1$

10.　　end if

11.　end for

12.　return A_D

算法 1 中,第 3 行与第 6 行 $F_{\text{rb}}()$ 为通过重构概率模型计算出重构概率分数 S;第 4 行 $t_{h_b} = F_{\text{EVT}}()$ 为利用重构误差模型的训练得到的重构概率得分 S 计算出最优阈值 t_{h_b}。第 7 – 第 10 行将重构概率得分与最优阈值相比较,若 $S_i < t_{h_b}$,则 x_i 被视为异常,i 时刻系统被判定为处于异常状态,反之为正常。

(1) 重构概率计算。重构概率计算主要以变分自编码器为主要框架,根据 VAE 的工作原理,准确获得数据分布信息是能准确重构数据的关键,所以必须充分学习监控指标的数据特征。本文将 1D-CNN、SGRU 和 SE 结构融合,作为数据特征提取网络结构,并以 VAE 模型作为主要框架来计算数据重构概率。重构概率计算网络结构如图 3 所示,对于输入的监控指标数据,由 1D-CNN 学习数据短期依赖特征,SGRU 从 1D-CNN 学习的特征中学习时间依赖性,捕获粗粒度特征,通过 Dense 层实现数据细粒度特征与粗粒度特征的融合,以得到带有监控指标数据特征的隐空间变量。VAE 模型通过带有数据特征的隐空间变量重构数据,得出重构概率。

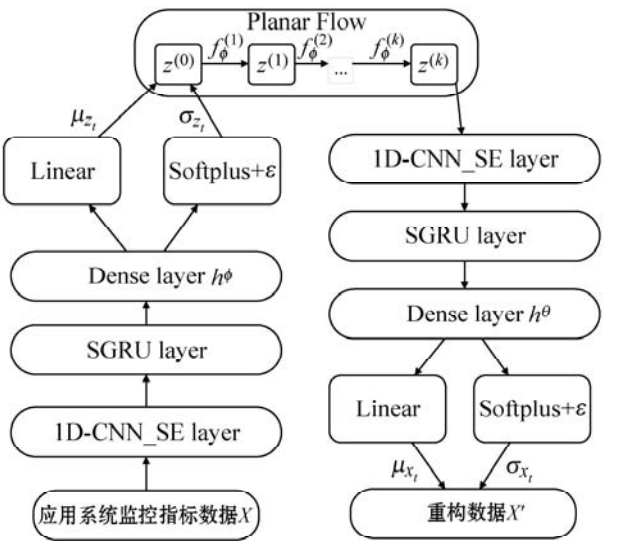


图 3　重构概率模型

由于监控指标数据具有相关性,且在时间维度上数据特征的重要程度不同,某些指标数据的特征重要程度大于其他指标[19],仅使用传统的 1D-CNN 网络会忽略学习指标数据的这一特点。本文引入 SE 网络结构,帮助 1D-CNN 网络建模特征相关性,从全局信息中识别出不同指标特征对于分类的重要程度。首先使用 1D-CNN 捕获数据的短期数据局部特征,再通过 SE[20]

网络结构建模全局信息中不同通道之间的相互依赖关系,根据全局信息来选择性地强调重要特征并抑制非重要特征。1D-CNN_SE 网络结构如图 4 所示。

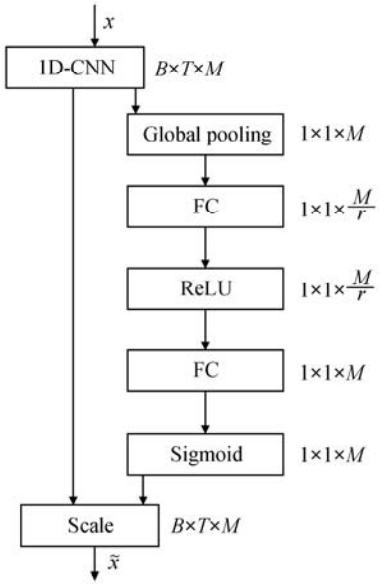


图 4 1D-CNN_SE 网络结构

将预处理后的数据 $\boldsymbol{x} \in \mathbf{R}^{(B\times T\times M)}$ 输入到 1D-CNN 中,为防止模型过拟合,加快模型训练速度,加入批量归一化(Batch Normalization,BN)与激活函数 ReLU,得到 1D-CNN 提取的局部数据特征 f。为学习数据之间的相关性,并且输出带有特征重要程度信息的特征,将 $\boldsymbol{f}$ 输入到 SE 网络结构中。

首先经过卷积算子 F_{tr} 对指标数据进行卷积运算,将输入 $\boldsymbol{f} \in \mathbf{R}^{B\times T\times M}$ 转化为向量矩阵 $\boldsymbol{U} = [\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_M]$:

$$\boldsymbol{u}_M = \boldsymbol{v}_M * \boldsymbol{f} = \sum_{s=1}^{M} \boldsymbol{v}_M^s * \boldsymbol{f}^s \tag{11}$$

式中:$\boldsymbol{U}$ 由 M 个大小为 $B \times T$ 的特征图组成;$\boldsymbol{u}_M$ 表示 $\boldsymbol{U}$ 中第 M 个二维矩阵,M 表示通道;二维空间核用 $\boldsymbol{v}_M^s$ 表示,$\boldsymbol{v}_M$ 的单个通道作用于 $\boldsymbol{X}$ 的相应通道。$\boldsymbol{u}_M$ 由所有通道的和产生,特征依赖与滤波器捕获的空间特性被保留在 $\boldsymbol{v}_M$ 中。

从各维指标特征中学习重要程度,需获得指标的全局空间信息,本文通过 global-pooling 操作得到 $\boldsymbol{U}$ 的全局信息 $\boldsymbol{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_M]$:

$$\boldsymbol{Z}_M = \frac{1}{B \times T} \sum_{i=1}^{B} \sum_{j=1}^{T} \boldsymbol{u}_M(i,j) \tag{12}$$

$\boldsymbol{Z}$ 是通过在空间维度 $B \times T$ 上转化 $\boldsymbol{U}$ 生成的;随后运用 ReLU 函数和 sigmoid 函数分别学习特征间非线性关系并归一化权重,即得到各维度指标带有特征重要程度权重的空间特征 $\tilde{\boldsymbol{x}} = [\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{x}}_2, \cdots, \tilde{\boldsymbol{x}}_M]$,计算过程如下:

$$\boldsymbol{s} = \mathrm{sigmoid}(\boldsymbol{W}_2 \mathrm{ReLU}(\boldsymbol{W}_1 \boldsymbol{Z})) \tag{13}$$

$$\tilde{\boldsymbol{x}}_M = F_{\mathrm{scale}}(\boldsymbol{u}_M, \boldsymbol{s}_M) = \boldsymbol{s}_M \cdot \boldsymbol{u}_M \tag{14}$$

为捕获指标数据的时间依赖粗粒度的特征,将带有指标空间特征的向量矩阵 $\tilde{\boldsymbol{x}}$ 输入到 SGRU 中得到数据时间依赖特征 $\boldsymbol{H} = [h_1, h_2, \cdots, h_t]$,随后将 1D-CNN_SE 与 SGRU 提取的特征映射到 Dense 层上,实现粗细粒度特征融合。

由于 VAE 模型假设隐变量符合高斯分布,但实际场景中,指标数据不一定符合简单的高斯分布,模型如果不能正确学习数据分布会导致模型不能拟合足够复杂的后验分布[21],受文献[13]启发,使用 Planar NF[22] 对 $q_\phi(\boldsymbol{z} \mid x)$ 生成的隐变量 $\boldsymbol{z}^{(0)}$ 在平面空间对假设符合高斯分布的隐变量进行可逆变换,变换为复杂灵活的接近真实分布的后验分布,如式(15)所示。

$$f^k(\boldsymbol{z}^{k-1}) = \boldsymbol{z}^{k-1} + \mu \tanh(\boldsymbol{w}^{\mathrm{T}} \boldsymbol{z}^{k-1} + \boldsymbol{b}) \tag{15}$$

VAE 根据 1D-CNN_SE、SGRU 捕获的数据特征重构数据,最小化损失函数得到最优参数 θ^*、ϕ^* 训练模型,VAE 损失函数定义为:

$$L(\phi, \theta; x) = \frac{1}{N} \sum_{i=1}^{N} (\log p_\theta(x_i \mid z_i)) - D_{\mathrm{KL}}(q_\phi(z_i \mid x_i) \parallel p_\theta(z_i)) \tag{16}$$

将重建概率 $S_i = \log(p_\theta(x'_i \mid z_{i-T+1,i}))$ 用来评估重建 x_i 的程度,重构概率分数越低,则越可能是异常。重构概率计算描述如算法 2 所示。

算法 2 重构概率计算算法

输入:监控指标数据 $\boldsymbol{X} \in \mathbf{R}^{(N\times M)}$。

输出:重构概率分数 S。

1. for $i = 1$ to N do
2. $\quad \boldsymbol{x} = normalize(\boldsymbol{X})$
3. $\quad \boldsymbol{x}_{i-T+1:i} \in \mathbf{R}^{T\times M} = slidingwindows(\boldsymbol{x})$
4. $\quad \mu_{z_0^i}, \delta_{z_0^i} = q_\phi(\boldsymbol{z}_0^i \mid \boldsymbol{x}_i)$
5. $\quad \boldsymbol{z}_i^{(0)} = \mu_{z_i} + \xi_i \sigma_{z_i}$
6. $\quad \boldsymbol{z}_i^K \leftarrow f^K(f^{K-1}(\cdots f^1(\boldsymbol{z}_i^0)))$
7. $\quad \boldsymbol{\mu}_{x'_i}, \boldsymbol{\delta}_{x'_i} = p_\theta(\boldsymbol{x}'_i \mid \boldsymbol{z}_i^K)$
8. $\quad S_i = \log(p_\theta(\boldsymbol{x}'_i \mid \boldsymbol{z}_{i-T+1,i})$
9. end for
10. return S

算法 2 中第 2、第 3 行为数据预处理,通过 min-max 归一化将数据归到(0,1)区间,使用滑动窗口将数据分成若干长度为 T 的子序列 $\boldsymbol{x}_{i-T+1:i}$,逐步输入到以 VAE 为框架的重构概率模型中;第 4、第 5 行通过 VAE 模型的编码器 $q_\phi(z \mid x)$ 将多元指标数据 $\boldsymbol{x}_i \in \mathbf{R}^{1\times M}$ 通过 1D-CNN_SE 与 GRU 网络的特征提取后,将其编码为隐空间变量 $\boldsymbol{z}$;第 6 行通过 Planar NF 将隐变量通过可逆变换,将 VAE 假设的近似后验分布变换为复杂灵活的接近真实分布的后验分布;第 7 行将接近真实分布

的隐变量 $\boldsymbol{z}^K$ 通过与编码器相同的神经网络结构重构出多元指标数据的分布 $\boldsymbol{x}'_i$;第 8 行将 $\boldsymbol{x}'_i$ 与训练好的重构概率模型生成的原始多元指标数据 $\boldsymbol{x}_i$,计算重构概率分数 S_i。

(2) 最优异常阈值计算。许多异常检测算法都假设测量数据遵循特定的分布(如高斯分布),这种假设不适用于应用系统工作环境,因为环境工作负载会随时间而变化[23]。而极值理论的特点,不用对数据的分布进行任何前提假设的情况下对极值概率进行计算。

而在本文方法中,异常数据的重构概率低于正常数据的重构概率,所以异常阈值需要通过分析异常出现在分布的低端来确定,根据上文得到的重构概率分数 $S=\{S_1,S_2\cdots,S_N\}$,将 GPD 函数进行修改,如式(17)所示。

$$\tilde{F}(y)=P(t_h-S_i>y\mid S_i<t_h)\sim\left(1+\frac{\gamma y}{\beta}\right)^{-\frac{1}{\gamma}} \quad (17)$$

式中:$i\in(1,N)$;γ、β 为 GPD 的形状和比例参数,在已知样本拟合为 GPD 分布的情况下,采用极大似然估计的参数估计方法确定最优 $\hat{\gamma}$ 和 $\hat{\beta}$;t_h 为初始阈值,按照经验设定为 98%[18]。通过式(10)计算最优阈值。其中 q 按照经验设定为 10^{-4}[18]。最优异常阈值计算如算法 3 所示。

算法 3　最优异常阈值计算算法

输入:重构概率分数 $S=\{S_1,S_2,\cdots,S_N\}$。

输出:最优异常阈值 t_{h_b}。

1. $t_h\leftarrow InitialThreshold(S_1,S_2,\cdots,S_N)$
2. for $i=1$ to N do
3. $\quad\tilde{F}(y)=P(t_h-S_i>y\mid S_i<t_h)$
4. end for
5. $\hat{\gamma},\hat{\beta}=\mathrm{MLE}(\tilde{F}(y))$
6. $t_{h_b}=t_h-\frac{\hat{\beta}}{\hat{\gamma}}\left(\left(\frac{qN}{N_{\mathrm{th}}}\right)^{-\hat{\gamma}}-1\right)$
7. Return t_{h_b}

算法 3 中,第 1 行首先初始化阈值;第 2 - 第 4 行根据 GPD 分布计算阈值方法学习重构概率分数得分的极值分布;第 5 行通过极大似然估计确定第 3 行函数的最佳参数;第 6 行根据 POT 方法计算出最佳阈值。

4　实验与结果分析

4.1　实验数据集

SMD 数据集是某互联网公司 5 周的监控数据的数据集,SOD 数据集通过操作系统和 WebLogic 服务器监控作为数据源,监控多个主机上的应用程序的活动状态的监控指标数据。数据集的基本信息如表 1 所示。

表 1　数据集基本信息

数据集	实体数	指标数	数据量	异常占比/%
SMD	28	38	1.5×10^6	4.16
SOD	20	231	7.5×10^6	37.7

主要监控指标如表 2 所示。

表 2　应用程序主要监控指标

指标类别	指标名称
Infrastructure	CPU usage/time
	Disk space usage
	Swap activity
	Physical memory usage
Database	Currently active connections
	Database connection delay
	Started/Failed/Successful connections
Middleware	total thread count
	Garbage collection activity
	Relative swap usage
	Database connection activity
	Transaction rollback/commit activity
Application	Response time
	Class loading/unloading activity
	Page view
	Missing data
	Relative open file descriptors

4.2　实验环境与参数设置

实验环境为 Intel(R)Core(TM)i7-7700 处理器,基本频率为 3.6 GHz,16 GB 内存,使用 CPU 训练,Windows 7 64 位操作系统。编程语言为 Python 3.6,深度学习框架为 TensorFlow。数据预处理中标准化步骤使用 min-max 归一化处理,训练过程中使用 Adam 优化器优化参数。实验超参数设置如表 3 所示。

表 3　实验超参数设置

参数	设置
Learning rate	10^{-3}
Epochs	50
Batch size	50
Planar NF	20
GRU hidden units	500

续表 3

参数	设置
CNN kernel-size	3,5,8
CNN filters	100,200,100
r	16

4.3 评价指标

异常检测可以被转化为二分类问题,本文设置精确率(Precision)、召回率(Recall)和 F1 值作为模型效果的评价指标。计算公式如下:

$$F_1 = \frac{2 \times P_{\text{recision}} \times R_{\text{ecall}}}{P_{\text{recision}} + R_{\text{ecall}}} \tag{18}$$

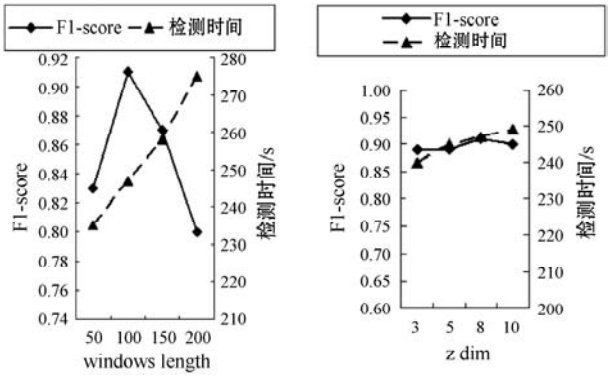
$$P_{\text{recision}} = \frac{T_P}{T_P + F_P} \tag{19}$$

$$R_{\text{ecall}} = \frac{T_P}{T_P + F_N} \tag{20}$$

式中:T_P(true positive)表示成功检测出的异常运行状态数量;F_P(false positive)表示正常状态被异常检测模型判断为异常的数量;F_N(false negative)表示异常状态被异常检测模型判断为正常的数量。

4.4 对比实验结果分析

(1) 模型关键参数对异常检测效果的影响。滑动窗口大小(windows length)与隐变量维度(z dim)的取值对本文异常检测效果起重要作用,且取值很大程度上取决于数据集。本文以 SMD 数据集为例,对以上两个参数的取值进行实验探究。实验结果如图 5 所示。

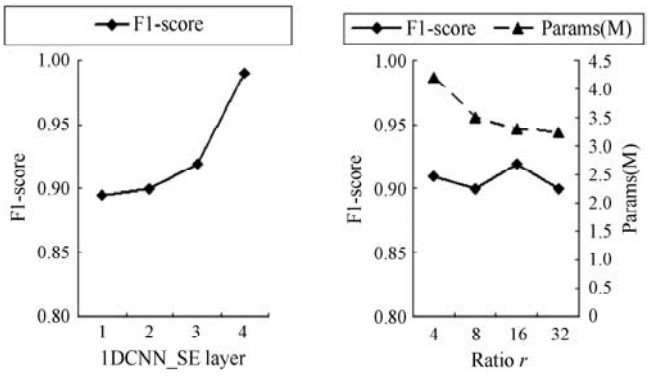


(a) 滑动窗口大小 (b) 隐变量空间维度

图 5 CS-VAE 关键参数对比实验

可以看出,在其他参数相同的情况下,滑动窗口大小取值为 100,隐变量空间维度大小取值为 8 时,异常检测效果与性能之比达到最佳。

1D-CNN 与 SE 网络作为本文新引入的网络结构,以 SMD 数据集为例,针对其关键参数的取值对异常检测结果的影响进行探究,主要包括 1D-CNN_SE 层数、SE 减速比 r,如图 6 所示。



(a) 1D-CNN_SE 层数 (b) SE 超参数 r 取值

图 6 1DCNN_SE 超参数对比实验

在图 6 中,通过实验发现 1D-CNN_SE 的层数取值为 3 层时 F1 值达到最好,再加一层导致模型训练发生过拟合,受全卷积网络结构的三层卷积核大小取值的启发,将本文的卷积核大小取值为 8、5、3,并根据滑动窗口大小确定卷积核数量为 100、200、100。SE 网络结构的超参数对于检测效果并无过大影响,主要作用体现在减少参数,从图 6 中得出 r 为 16 时能有效减少训练参数。

(2) 本文方法组成部分对异常检测效果的影响。为验证模型的各组成部分对异常检测结果的影响,以 SMD 数据集为例,针对本文方法与以下五个模型进行消融对比实验,通过 F1 值评定对比结果。实验结果如表 4 所示。

表 4 消融实验对比结果

消融模型	F1-Score
no 1D-CNN_SE	0.88
no SE	0.87
CS-VAE(LSTM)	0.74
no SSM	0.83
no Planar NF	0.80
CS-VAE	0.92

从消融实验结果可以得出,1DCNN_SE 结合 SGRU,一定程度上融合了粗细粒度,并从全局信息中获得特征重要程度,这是因为异常时刻指标特征重要程度大于正常时刻,从而提高分类效果,在 F1 指标上本文方法最优;而使用 LSTM 替换 GRU 进行实验的结果明显变差,可能是因为 GRU 有较少的参数和更简单的结构,因此更容易训练;SSM 使得隐变量空间保留了数据在神经网络中提取的特征,使得 VAE 模型更好地重构数据,提高检测效果;而 Planar NF 结构捕获具有复杂分布的数据模式,帮助隐变量空间更好地保留数据复

杂分布特点,提高检测效果。

(3) 本文方法与其他方法对比实验结果分析。为验证本文方法的有效性,选择了多个应用广泛的异常检测方法与本文进行比较,分别是文献[11]提出的 DAGMM 方法、文献[13]提出的 OmniAnomaly、文献[24]提出的局部异常因子(Local Outlier Factor,LOF)方法、文献[25]提出的孤立森林(Isolation Forest,IForest)方法。LOF 是基于密度的离群点检测方法,该算法会给数据集中的每个点计算一个离群因子 LOF,通过判断 LOF 是否接近于 1 来判定是否是异常点。若 LOF 远大于 1,则认为是异常点,接近于 1,则是正常点。IForest 将异常看作在数据空间里面,分布稀疏的区域的数据,数据发生在此区域的概率很低,落在这些区域里的数据是异常的。分别在 SMD 与 SOD 数据集上进行实验,实验结果如表 5 和表 6 所示。

表 5　不同方法在 SMD 数据集上的对比结果

算法	Precision	Recall	F1
LOF	0.56	0.55	0.56
IForest	0.78	0.77	0.78
DAGMM	0.75	0.63	0.69
OmniAnomaly	0.83	0.94	0.88
CS-VAE	0.89	0.93	0.92

表 6　不同方法在 SOD 数据集上的对比结果

算法	Precision	Recall	F1
LOF	0.47	0.60	0.52
IForest	0.52	0.52	0.52
DAGMM	0.75	0.51	0.63
OmniAnomaly	0.74	0.78	0.76
CS-VAE	0.79	0.73	0.76

从表 5 可以看出,在 SMD 数据集上,本文方法的 Precision、Recall 和 F1 值优于 LOF、IForest 和 DAGMM 方法,相比于 OmniAnomaly 方法,F1 值综合评价指标最优提升了 0.04。从表 6 可以看出,在 SOD 数据集上,本文方法与 OmniAnomaly 方法 F1 值相当,但本文方法的 Precision 值高于 OmniAnomaly 方法,表明本文提出的异常检测方法优于目前已提出的先进方法。这是因为 LOF 和 IForest 方法未能考虑到数据的时间依赖性,也不能很好地学习到数据间非线性的复杂相关性,导致异常检测的效果不好。DAGMM 方法将数据看作是多变量数据,未考虑时间依赖性,且通过固定阈值判定异常,导致召回率相差较大。OmniAnomaly 方法考虑到了监控指标的时间依赖性和随机性,所以比 LOF、IForest 和 DAGMM 方法效果更好,但是未考虑数据相关性和特征在不同时刻的重要程度的不同,导致特征刻画数据不全面,效果不如本文方法。对比实验也证明了本文模型更全面地学习了数据的特征,有效提高了异常检测效果。

4.5　实验过程可视化

CS-VAE 对正常数据进行训练,学习其数据特点。由于异常数据具有异于正常数据特征的特点,在向 VAE 的编码器输入异常样本时,生成器重构异常样本会产生较低的重构概率。这可以很好地体现系统各种异常产生的指标突变,从而准确检测异常。

为了直观地观察这一点,通过应用系统的一个具体异常实例说明 CS-VAE 方法的实际有效性。

图 7 描述了应用系统由于数据库资源发生异常,导致服务响应时长异常的实例。图 7 显示受数据库异常故障传播影响,应用系统指标,包括 CPU 利用率、磁盘负载、数据库读取操作次数、数据库响应时间、数据库缓冲池利用率、JDBC 线程数等指标数据在 500 ~ 1 000 时间戳内发生不同于正常运行状况(如 0 ~ 500、1 000 ~ 1 500 时间戳所示)的数值突变。本文提出的异常检测方法 CS-VAE 通过神经网络结构有效地检测出系统在运行过程中发生的指标突变异常,并通过 POT 极值模型确定阈值,实现端到端异常检测,减少异常检测对人力与运维经验的依赖。

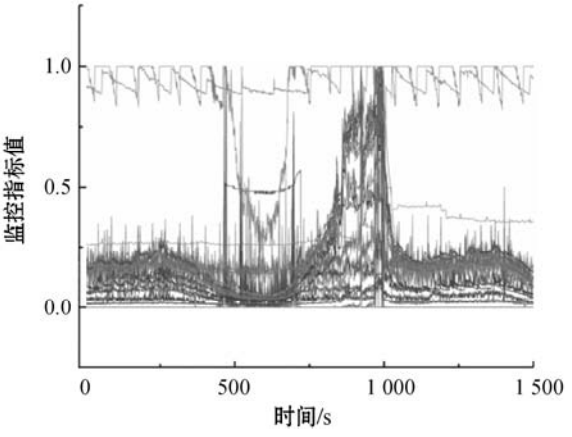


图 7　应用系统异常实例

将正常运行状态下的数据(如 0 ~ 500,1 000 ~ 1 500 时间内数据)输入到 CS-VAE 中,对神经网络模型进行训练,学习正常状态下各指标的规律变化,并得到最优阈值。将发生异常的监控指标数据(如图 7 中 500 ~ 1 000 时间内数据所示)输入到训练好的模型中进行异常检测,在通过重构概率计算模型对发生异常数据进行重构后,将异常部分指标重构成符合高斯分布的正常指标,如图 8 所示。此时正常时刻的各维监控指标重构概率之和是符合常规的,如图 9 中的 0 ~

500 与 1 000 ~ 1 500 时间戳内的重构概率分数取值所示,但由于异常数据是难重构的,导致各维异常数据的重构概率分数之和是明显低于正常时刻的,如图 9 中 500 ~ 1 000 时间戳内的重构概率分数所示。

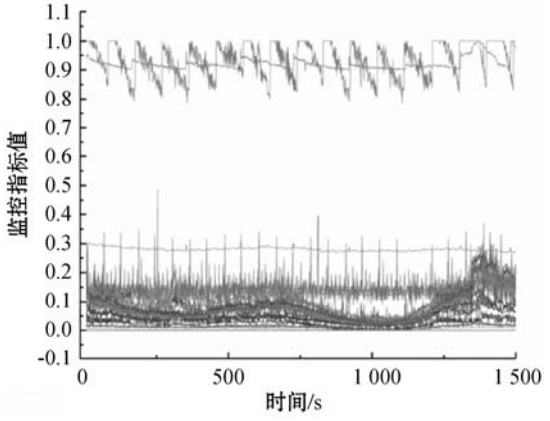


图 8 重构后的指标数据

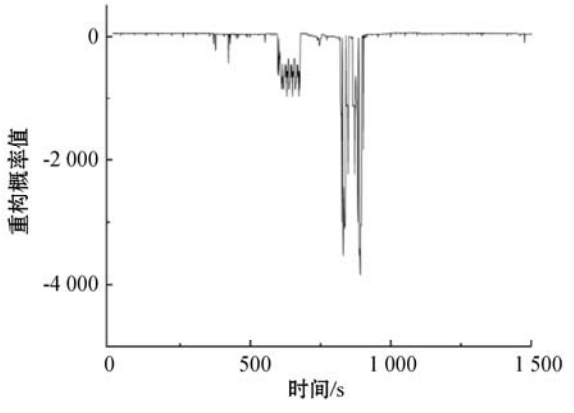


图 9 指标数据重构概率

将训练时利用正常数据计算的值作为最优异常阈值,正常数据的重构概率之和是高于最优异常阈值的,如图 10 中 0 ~ 500 和 1 000 ~ 1 500 时间戳取值所示,而异常数据的重构概率之和低于最优异常阈值,如图 10 中时间戳 500 ~ 1 000 所示,从而得出在 500 ~ 1 000 时间戳时,系统的运行状态发生异常。

通过对以上实例进行异常检测,可以看出本文方法能够准确检测出系统在运行状态中产生的异常。

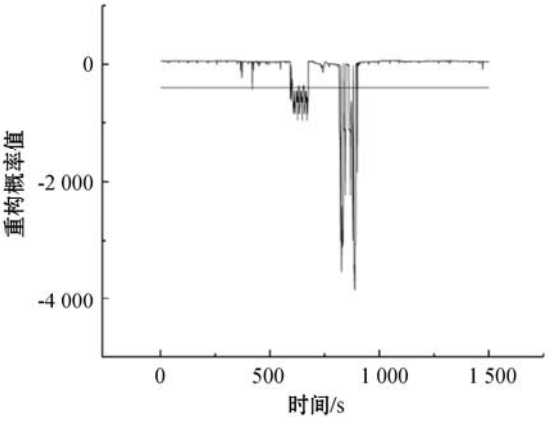


图 10 指标重构概率与阈值

5 结 语

本文提出的一种融合监控指标数据多种特征的面向应用系统监控指标的异常检测方法 CS-VAE。从数据的相关性和特征重要程度、时间依赖性等方面全面地学习了监控数据的特征,实现了粗细粒度特征结合,有效地提高了异常检测效果,实验结果表明,CS-VAE 方法在准确性方面优于现有的大多数异常检测方法。但本文方法在效率方面表现不突出,且仍需要一部分异常标签,未来需要在这两方面改进,得到更符合真实运维环境要求的异常检测方法。

参 考 文 献

[1] 朱海麒,姜峰. 人工智能时代面向运维数据的异常检测技术研究与分析[J]. 信息网络安全,2019(11):24 - 35.

[2] Lu W, Ghorbani A. Network anomaly detection based on wavelet analysis[J]. EURASIP Journal on Advances in Signal Processing,2009,2009:1 - 16.

[3] Yaacob A, Tan I, Chien S, et al. ARIMA based network anomaly detection[C]//2010 Second International Conference on Communication Software and Networks,2010:205 - 209.

[4] Hochenbaum J, Vallis O, Kejariwal A. Automatic anomaly detection in the cloud via statistical learning[EB]. arXiv: 1704.07706,2017.

[5] Liu F, Ting K, Zhou Z. Isolation forest-based anomaly detection[J]. ACM Transactions on Knowledge Discovery from Data,2012,6(1):1 - 39.

[6] Liu D, Zhao Y, Xu H, et al. Opprentice: Towards practical and automatic anomaly detection through machine learning [C]//2015 Internet Measurement Conference,2015:211 - 224.

[7] Ma J, Perkins S. Time-series novelty detection using one-class support vector machines[C]//International Joint Conference on Neural Networks,2003:1741 - 1745.

[8] Ester M, Kriegel H, Xu X, et al. A density-based algorithm for discovering clusters in large spatial databases with noise [C]//2nd International Conference on Knowledge Discovery and Data Mining,1996:226 - 231.

[9] Kingm D, Welling M. Auto-encoding variational bayes [EB]. arXiv:1312.6114,2013.

[10] Li D, Chen D, Shi L, et al. MAD-GAN: Multi-variate anomaly detection for time series data with generative adversarial networks[EB]. arXiv:1901.04997,2019.

[11] Zong B, Song Q, Min M, et al. Deep autoencoding Gaussian mixture model for unsupervised anomaly detection[C]//

International Conference on Learning Representations,2018:1 - 19.
[12] 任浩,屈剑锋. 深度学习在故障诊断领域中的研究现状与挑战[J]. 控制与决策,2017,32(8):1345 - 1358.
[13] Su Y, Zhao Y, Niu C, et al. Robust anomaly detection for multivariate time series through stochastic recurrent neural network[C]//25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2019: 2828 - 2837.
[14] Khoshnevisan F, Fan Z. RSM-GAN: A convolutional re-current GAN for anomaly detection in contaminated seasonal multivariate time series[EB]. arXiv:1911.07104,2019.
[15] Huch F, Golagha M, Petrovska A, et al. Machine learning-based run-time anomaly detection in software systems: An industrial evaluation[C]//2018 IEEE Workshop on Machine Learning Techniques for Software Quality Evaluation,2018: 13 - 18.
[16] Cho K, Merrienboer B, Gulcehre C. Learning phrase representations using RNN encoder-decoder for statistical machine translation[EB]. arXiv:1406.1078,2014.
[17] Fraccaro M, Sønderby S, Paquet U, et al. Sequential neural models with stochastic layers[C]//30th International Conference on Neural Information Processing Systems, 2016: 2207 - 2215.
[18] Siffffer A, Fouque P, Termier A, et al. Anomaly detection in streams with extreme value theory[C]//23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,2017:1067 - 1075.
[19] 孟恒宇,李元祥. 基于 Transformer 重建的时序数据异常检测与关系提取[J]. 计算机工程,2021,47(2):69 - 76.
[20] Hu J, Shen L, Albanie S, et al. Squeeze-and-excitation networks[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition,2017:7132 - 7141.
[21] Rezends D, Mohamed S. Variational inference with normalizing flows[EB]. arXiv:1505.05770,2015.
[22] Papamakarios G, Nalisnick E, Rezende D, et al. Normalizing flows for probabilistic modeling and inference[EB]. arXiv:1912.02762,2019.
[23] Lin J, Zhang Q, Bannazadeh H, et al. Automated anomaly detection and root cause analysis in virtualized cloud infrastructures[C]//NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium,2016:550 - 556.
[24] Breuning M, Kriegel H, Ng R, et al. LOF: Identifying density-based local outliers[J]. ACM SIGMOD Record,2000, 29(2):93 - 104.
[25] Liu F, Ting K, Zhou Z. Isolation-based anomaly detection [J]. ACM Transactions on Knowledge Discovery from Data, 2012,6(1):1 - 39.

(上接第 21 页)

参 考 文 献

[1] yacer 亚册. HDLC-ATC 空管数据通信服务器用户手册[EB/OL]. (2020 - 10 - 21)[2021 - 02 - 21]. http://www.yacer.cn/Release/Products/HDLC-ATC/HDLC-ATC_用户手册.pdf.
[2] Nagel C. C#高级编程[M]. 北京:清华大学出版社,2019: 550 - 557.
[3] 刘晓天. 基于 TCP 协议的网络应用设计与开发[J]. 山东农业工程学院学报,2017,34(8):157 - 158.
[4] 刘帅. S 模式多雷达数据处理系统设计与实现[D]. 西安:西安电子科技大学,2017.
[5] 陈晓伟. ADS-B 数据格式解析[J]. 科技经济市场,2020(10):15 - 16.
[6] 刘邦强. 基于 C#的 ADS-B 数据模拟系统的设计与实现[J]. 现代计算机,2020(2):86 - 89.
[7] 易凡. 基于 C#的 ADS-B 测试平台的设计与实现[J]. 机电工程技术,2019,48(5):137 - 139.
[8] 任登国. 乌鲁木齐国产一、二次雷达数据格式[J]. 科技创新与应用,2019(24):61 - 63.
[9] 赵文斌. ASTERIX CAT048 数据格式分析[J]. 中国新技术新产品,2018(1):26 - 27.
[10] 胡建. 二次雷达探测仿真软件设计与实现[D]. 成都:西南交通大学,2017.
[11] 周日红. 空管雷达监视数据格式浅析[J]. 电子世界,2017(22):56 - 57.
[12] 高兴. INDRA 雷达监控管理系统的设计与实现[D]. 广州:广东工业大学,2019.
[13] 刘燕. 基于 ADS-B 监视数据处理技术的研究与实现[D]. 南京:东南大学,2018.
[14] Eurocontrol. EUROCONTROL specification for surveillance data exchange ASTERIX part 4 category 048 Monoradar target reports[EB/OL]. (2020 - 06 - 18)[2021 - 02 - 21]. https://www.eurocontrol.int.
[15] 郭进祥. 基于深度学习的机场场面飞机检测跟踪系统设计与实现[D]. 银川:宁夏大学,2019.
[16] 许文君. 空管自动化系统及数据融合方法研究[D]. 南京:南京邮电大学,2019.
[17] 刘云丰,廖盈庭,刘书博. 基于 Python 的 Asterix Cat 021 数据格式解析分析与实现[J]. 科技与创新,2019(14): 46 - 47.
[18] 杜广正. 一种二次雷达原始数据回放分析系统的应用研究[D]. 西安:西安电子科技大学,2018.