GRD:基于 GNN 和扩散模型的多变量时序数据异常检测算法

邸钦渤[1],陈劭力[2],时良仁[1*]

(1.上海交通大学电子信息与电气工程学院,上海 200240;2.华讯网络系统有限公司,上海 200127)

摘 要: 随着多变量时序数据在各行业中的广泛应用,开发有效的异常检测方法对于保障系统的稳定运行和安全性变得极为关键,由于多变量时序数据内在的复杂性和动态变化特性,对异常检测算法提出了更高的要求。针对现有异常检测方法在处理含有复杂变量关系的高维数据时存在效率不足的问题,提出一种基于图神经网络(GNN)与扩散模型的多变量时序数据异常检测算法 GRD。通过节点嵌入和图结构学习,GRD 算法能有效地捕捉和表示变量间的复杂关系,并通过门控循环单元(GRU)和去噪扩散概率模型(DDPM)进一步提取特征,实现了对异常数据的高精度检测。在以往的实验评估中,大多数算法在评分前会采用点调整(PA)评估协议,该协议会严重高估算法的检测能力。为了更准确地评估算法性能,采用新的评估协议和评价指标。实验结果表明,GRD 算法在 3 个公开数据集上的 F1@k 指标分别是 0.741 4、0.801 7、0.767 1,性能优于现有方法。特别是在高维数据处理方面,GRD 算法展现出显著优势,证明了其在现实场景的异常检测应用中的实用性和鲁棒性。

关键词: 多变量时序数据;异常检测;图神经网络;扩散模型;评估协议

源代码链接: https://github.com/dqb35/GRD

中图分类号: TP183 **文献标志码:** A **DOI:** 10.19678/j.issn.1000-3428.0069780

GRD:Anomaly Detection Algorithm for Multivariate Time Series Data Based on GNN and Diffusion Model

DI Qinbo[1], CHEN Shaoli[2], SHI Liangren[1*]

(1.School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China;
2.ECCOM Network System Co., Ltd., Shanghai 200127, China)

【Abstract】 As multivariate time series data become increasingly prevalent across various industries, anomaly detection methods that can ensure the stable operation and security of systems have become crucial. Owing to the inherent complexity and dynamic nature of multivariate time series data, higher demands are placed on anomaly detection algorithms. To address the inefficiencies of existing anomaly detection methods in processing high-dimensional data with complex variable relations, this study proposes an anomaly detection algorithm for multivariate time series data, based on Graph Neural Networks (GNNs) and a diffusion model, named GRD. By leveraging node embedding and graph structure learning, GRD algorithm proficiently captures the relations between variables and refines features through a Gated Recurrent Unit (GRU) and a Denoising Diffusion Probabilistic Model (DDPM), thereby facilitating precise anomaly detection. Traditional assessment methods often employ a Point-Adjustment (PA) protocol that involves pre-scoring, substantially overestimating an algorithm's capability. To reflect model performance realistically, this work adopts a new evaluation protocol along with new metrics. The GRD algorithm demonstrates F1@k scores of 0.741 4, 0.801 7, and 0.767 1 on three public datasets. These results indicate that GRD algorithm consistently outperforms existing methods, with notable advantages in the processing of high-dimensional data, thereby underscoring its practicality and robustness in real-world anomaly detection applications.

【Key words】 multivariate time series data; anomaly detection; Graph Neural Network (GNN); diffusion model; evaluation protocol

0 引言

在现代社会中,随着物联网技术和智能设备的不断普及,多变量时序数据在各个领域中的应用越来越广泛。这些数据不仅包含时间序列的动态变化,还涉及多个变量之间的复杂关系[1-4]。由于数据的高维性和复杂性,对其进行有效异常检测具有重要意义,可应用于预测系统故障、优化用户体验和防范金融风险等领域。

近年来,许多研究工作致力于多变量时序数据

收稿日期: 2024-04-24 **修回日期:** 2024-07-23

通信作者 E-mail: * leonsong@sjtu.edu.cn

的异常检测研究。传统的统计模型和机器学习算法在处理低维数据时表现良好,但在面对高维数据时往往受限于计算复杂度和模型表达能力。随着深度学习技术的发展,一些基于深度学习的方法被提出,如基于自编码器的重构方法[5]和基于循环神经网络(RNN)的预测方法[6]等,这些方法在一定程度上提高了异常检测的性能。然而,这些方法往往忽略了数据中变量间的关系,导致在实际处理具有复杂关系的多变量时序数据时效果不佳。此外,许多关于时序数据异常检测的研究报告在相关基准数据集上获得了很高的 F1 值,但在实际的部署应用中却不尽如人意,原因是大多数研究在指标计算前都会采用点调整(PA)的评估协议。已经有研究证明[7-8]:该方案会大大高估模型的能力,无法反映真实的建模性能。

图神经网络(GNN)[9]和扩散模型[10]在数据分析领域的兴起为多变量时序数据异常检测提供了新的视角和方法。因此,本文提出一个新的多变量时序数据异常检测算法 GRD,其能够充分考虑多变量时序数据的时空特征和变量间关系。GRD 算法首先利用 GNN 建模数据中的变量间关系,然后通过门控循环单元(GRU)[11]捕捉时序信息,最后通过扩散模型进一步提取特征,从而实现对异常的准确检测。在模型评估方面,本文采用新的评估协议 PA%k[7]来应用于现有的指标。为了促进研究的复现和进一步发展,GRD 算法的代码已经开源。相关代码可以在 https://github.com/dqb35/GRD 上获取。

综上所述,本文的主要贡献包括:

1)将 GNN 与扩散模型相结合提出一种新算法——GRD。该算法能够捕捉多变量时序数据的时空关系,并通过扩散模型进一步平滑异常数据,从而显著提升异常检测的性能。

2)在 3 个公开数据集上进行了实验,并采用 PA%k 协议来应用于现有的指标。研究结果表明,GRD 算法比其他基线方法异常检测准确率更高,特别是在处理高维数据时展现了显著的优势。

3)通过实验分析研究得出:GRD 算法不仅可以通过图表示学习部分提供模型的可解释性,还能通过扩散模型部分进一步平滑异常数据,从而得到更准确的检测结果。

1 相关研究

根据多变量时序数据异常检测方法的主要特点,本文将其分为 2 类:第 1 类方法通过一些传统的机器学习算法对时序数据进行分析;第 2 类方法使用深度学习算法将时序数据作为一个统一的实体进行建模。对于第 1 类方法,本文将其细化为分类模型、基于距离的模型和统计模型。对于第 2 类方法,本文将其细分为基于预测的模型、基于重构的模型和混合模型。除了这 2 类方法之外,本文还会额外说明几种新兴的时序数据异常检测方法。

1.1 机器学习算法

1.1.1 分类模型

在时序数据异常检测的发展初期,研究者逐渐将传统的机器学习算法应用于该领域,如一类支持向量机(1-SVM)、支持向量数据描述(SVDD)、自回归集成移动平均(ARIMA)、假设检验等。此类方法分为监督式和半监督式,因此大量的标注工作是不可避免的。

1.1.2 基于距离的模型

基于距离的模型是一种常见的无监督异常检测模型。文献[12]使用基于密度的噪声应用空间聚类(DBSCAN)算法对时序数据进行聚类。与 k-means 聚类相比,DBSCAN 不需要指定数据集中的聚类个数 k,也可以避免 k-means 聚类通常只在球形分布数据集分类中表现良好的问题。

1.1.3 统计模型

统计模型是实际生产环境中应用最广泛的异常检测模型。文献[13]利用 ARIMA 模型解决交通时间序列预测和异常检测问题。虽然 ARIMA 在处理时间序列数据方面的性能已经被证明是有效并且优秀的,但是有一个严格的要求,即序列必须是平稳或差分后平稳的。但是,实际场景中的大多数时序序列都是非平稳的,这使得该算法很难获得好的结果。文献[14]利用截断奇异值分解正则化与 L 曲线优化(TSVD+L)方法对智能电表进行异常检测。然而,这些方法都对原始数据进行了某种强假设,如高斯分布、线性关系等,与许多实际情况不兼容。

1.2 深度学习算法

虽然基于机器学习的异常检测算法已经得到了实现,但在数据层面仍有一些特殊的要求,这使得其需要在各种实际生产环境中作出相应的调整。与传统的机器学习算法相比,深度学习网络可以更好地进行扩展,更容易适应不同的领域和应用,更适用于多变量时序数据异常检测的多背景场景。

1.2.1 基于预测的模型

大多数的多变量时序数据都是有时间周期的,因此通常使用基于预测的异常检测模型。它们大多是基于预测误差来检测异常。文献[15]利用堆叠的

长短期记忆(LSTM)网络反向预测 d 步时间序列，通过计算预测误差检测异常。文献[16]使用卷积神经网络(CNN)作为其预测模型，从给定的时间序列窗口中预测下一个时间戳。文献[17]采用标准正态序列训练时间卷积网络(TCN)，并预测多个时间步长的趋势。

1.2.2 基于重建的模型

基于重建的模型主要包含编码器-解码器架构网络，通常通过基于许多潜在变量重建原始输入来学习整个时间序列的表示，这本质上是一种数据压缩和降噪算法。通过重构时间序列与原始时间序列的整体误差和局部误差判断异常。简单地说，只要输出的时间序列的局部信息与原始时间序列不一致，就有理由认为原始时间序列存在异常。文献[18]提出了一种基于 LSTM 的编码器-解码器方案，该方案学习重构正态序列，利用重构误差检测异常。文献[19]设计了一种基于变分自编码器(VAE)的无监督多变量时序数据异常检测方法——Dount，最终使 VAE 实现了与自编码器相同的功能。文献[1]在考虑多变量时间序列的时间依赖性和随机性的同时通过学习潜在表示将 GRU 和 VAE 结合起来，在该方法中重建概率较低的模式将被视为异常。

如上所述，这些算法更适合于建模时间序列的时间关系，而忽略了空间依赖性。

1.2.3 混合模型

为了克服以往方法所面临的问题，研究者提出了一些更复杂的混合模型，通过组合多个深度网络或设计其他训练策略来提取更大范围的特征并获得全尺寸时间序列表示，从而获得更好的检测结果。文献[20]设计一种将自编码器与 TCN 相结合的无监督异常检测方法，旨在同时提取时间特征和不完整的空间局部特征。文献[21]利用自编码器和对抗训练策略将多变量时序数据转换成相关图，以获得更广泛的时间序列特征表示。尽管这些方法提升了异常检测性能，但它们需要消耗更多计算资源，且所采用的图结构无法准确表征多变量时序数据间的实际相关性。

GNN 是一种新兴的深度学习方法，适用于处理具有图结构的数据。在多变量时序数据异常检测中，GNN 可以用于捕捉变量之间的复杂关系。文献[9]和文献[22]都采用完全图形式对时间序列进行建模，这在很大程度上避免了数据图结构未知的问题并且可全面捕捉变量之间的关系，但同时引入了计算复杂度增加和噪声干扰等问题。

1.3 新兴算法

1.3.1 基于大模型的算法

传统的时间序列模型是针对特定任务的，具有单一功能和有限的泛化能力。最近，大型语言基础模型展现了它们优越的跨任务可转移性、零样本/少样本学习能力以及决策解释性。这一成果激发了研究者探索基础模型来解决多个时间序列分析问题的兴趣。主要的研究方向包括从头开始预训练时间序列的基础模型[23-24]以及适应大型语言基础模型以处理时间序列[25-26]。这两者都致力于发展一个统一、高度泛化、易于理解的时间序列分析模型。

1.3.2 基于扩散模型的算法

去噪扩散概率模型(DDPM)[10]已发展为一类强大的生成模型，在图像合成[27-28]、视频生成[29]、文本-语音转换[30]以及可控文本生成[31]等多种模态和任务中均取得了显著成果。

扩散模型已经在图像的异常检测[32-33]中取得了不错的效果。异常分值是首先在图像中加入噪声，然后去噪并测量原始图像与重建图像之间的差异得到的。DDPM 也被用于时间序列预测[34]。受此启发，PINTILIE 等[35]也尝试将扩散模型应用于多变量时序数据异常检测，但是在实际数据集上的效果不理想，有关该领域的研究还需新的进展，本文正是在此基础上进行的进一步研究。

2 GRD 算法设计

2.1 问题描述

时间序列数据包含连续的观测值，这些观测值通常在等时间戳跨度上收集[36]。本文更多关注的是多变量的时间序列数据 $\boldsymbol{X}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_T\}$，其中，$T$ 为 $\boldsymbol{X}$ 的时序长度，观测值 $\boldsymbol{x}_t\in\mathbb{R}^M$ 是 t 时刻的一个 M 维的向量，$\boldsymbol{x}_t=[x_t^1,x_t^2,\cdots,x_t^M]$，所以 $\boldsymbol{X}\in\mathbb{R}^{M\times T}$。

对于多变量时间序列异常检测，目标是确定观测值是否异常。对于时间序列建模，历史值有助于理解当前数据。与此同时，为了提高计算效率和减少内存使用，本文将上面表示的时序数据归一化并分成一系列的大小相等、长度为 w 的无重叠窗口：

$$\boldsymbol{W}_1=[\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_w]$$

$$\boldsymbol{W}_2=[\boldsymbol{x}_{w+1},\boldsymbol{x}_{w+2},\cdots,\boldsymbol{x}_{2w}]$$

$$\vdots$$

$$\boldsymbol{W}_l=[\boldsymbol{x}_{(l-1)w+1},\boldsymbol{x}_{(l-1)w+2},\cdots,\boldsymbol{x}_{lw}]$$

式中：$l=[T/w]$为窗口总数。

2.2 算法整体结构

GRD 算法的整体结构如图 1 所示。将经过预

处理后的数据送入网络，算法首先会根据数据的形状进行节点嵌入（Node Embedding）学习[37]，并且根据嵌入向量计算得到图的边索引（edge index）以及边权重（edge weight），将这些信息以及数据送入一个特定的 GNN（ARMAConv[38]）进行节点信息更新；然后通过 GRU 进一步捕捉时间步长信息，将更新后的数据信息通过基于重建的模块得到第一次重建后的数据；最后将第一次重建后的数据再经过 DDPM 进行第二次重建得到最终结果。

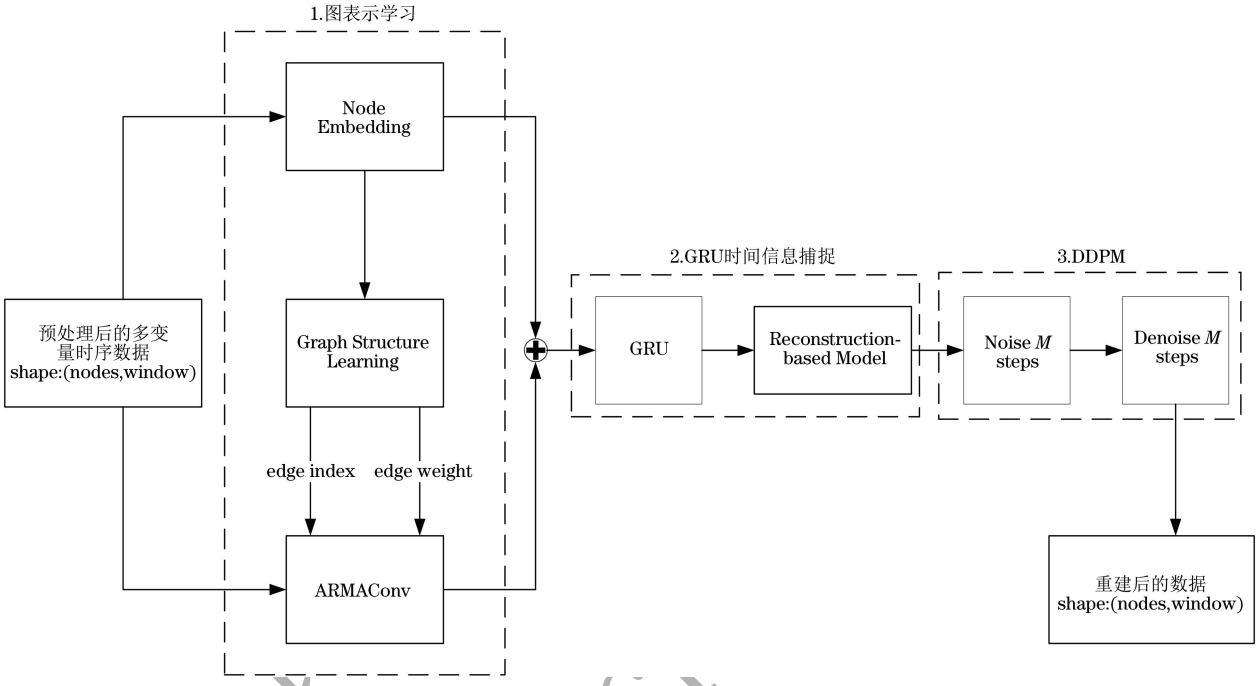

图 1 GRD 算法整体结构

Fig. 1 Overall structure of GRD algorithm

2.3 图表示学习

算法根据输入的数据学习节点嵌入向量以及图的结构，并将其送入特定的神经网络进行节点信息更新。

2.3.1 节点嵌入

在多变量时间序列数据中，不同变量通道的数据可能具有不同的特性，而这些特性可能以复杂的方式相互关联。因此，在理想情况下，本文将以一种灵活的方式表示每个通道的变量数据，以多维方式捕捉其行为背后的不同因素。

因此，本文通过为每个通道变量引入一个表示其特征的嵌入向量 $\hat{\boldsymbol{v}}_i \in \mathbb{R}^d, i \in \{1,2,\cdots,M\}$，其中，$d$ 为嵌入维度（超参数），将这些嵌入向量随机初始化，然后与模型的其余部分一起训练。

这些嵌入向量之间的相似性表示行为的相似性，因此具有相似嵌入值的通道变量通常具有较高的关联关系。

2.3.2 图结构学习

GRD 算法的一个主要目标是以图结构[39]的形式学习多变量时序数据之间的关系。为此，本文将使用一个有向图，节点表示一个通道变量的数据，边表示它们之间的依赖关系。使用有向图是因为不同通道变量之间的依赖模式不需要对称。本文使用边索引来表示这个有向图，边索引其实是一个两行的矩阵，其中第一行表示起始节点，第二行表示指向节点。

边权重是通过节点嵌入之间的余弦相似度[40]计算得到的。对于节点 i 和节点 j，边权重 e_{ij} 的计算公式如下：

$$e_{ij}=\frac{\hat{\boldsymbol{v}}_i^{\mathrm{T}}\hat{\boldsymbol{v}}_j}{\|\hat{\boldsymbol{v}}_i\|\cdot\|\hat{\boldsymbol{v}}_j\|},\ j\in\{1,2,\cdots,M\}\backslash\{i\} \tag{1}$$

边索引的确定方法是对所有节点对的权重进行排序，并选择权重最高的 Top-k 条边。

2.3.3 ARMAConv

GRD 算法的节点信息更新部分选择使用 ARMAConv，ARMAConv 是 GNN 中的卷积层。这种类型的层结合了自回归（AR）模型和移动平均（MA）模型，两者在时间序列分析中是常见的工具。在 GNN 的背景下，ARMAConv 层可以更好地捕捉图数据中的复杂模式，将数据以及前面计算得到的边索引和边权重输入该卷积层进行信息更新。

AR 部分模仿时间序列分析中的 AR 模型，它关注历史信息的影响。在图结构中，这意味着节点的特征不仅依赖于自身的历史特征，还依赖于其邻居节点的特征。对于图中的每个节点 v，AR 模型的每个层 k 的更新可以表示如下：

$$H_v^{(k)}=\sigma\Big(\sum_{u\in\mathcal{N}(v)}\frac{1}{c_{uv}}\boldsymbol{W}^{(k)}H_u^{(k-1)}+B^{(k)}\Big) \tag{2}$$

式中：$H_v^{(k)}$ 是节点 v 在第 k 层的隐藏状态；$\mathcal{N}(v)$ 是节点 v 的邻居；c_{uv} 是归一化常数，可由边权重系数计算得到；$\boldsymbol{W}^{(k)}$ 是 AR 权重矩阵；$B^{(k)}$ 是偏置项；σ 是非线性激活函数。

MA 部分引入额外的随机性，在每层中引入新的信息，并提高模型对输入数据中噪声的鲁棒性。对于节点 v，每个层 k 的 MA 更新可以表示如下：

$$M_v^{(k)}=\sigma\Big(\sum_{u\in\mathcal{N}(v)}\frac{1}{c_{uv}}\widetilde{\boldsymbol{W}}^{(k)}M_u^{(k-1)}+\widetilde{B}^{(k)}\Big) \tag{3}$$

式中：$M_v^{(k)}$ 是节点 v 在第 k 层的 MA 隐藏状态；$\widetilde{\boldsymbol{W}}^{(k)}$ 和 $\widetilde{B}^{(k)}$ 分别是 MA 权重矩阵和偏置项。

为了提高模型的表示能力，ARMAConv 通常堆叠多个这样的 AR 和 MA 计算单元，并将它们的输出组合起来。如果有 S 个堆叠，那么节点 v 的最终特征可以通过叠加每个堆叠的输出来得到：

$$H_v=\frac{1}{S}\sum_{s=1}^{S}(H_v^{(s)}+M_v^{(s)}) \tag{4}$$

式中：S 是堆叠数量；$H_v^{(s)}$ 和 $M_v^{(s)}$ 分别是 AR 和 MA 部分的最终输出。

2.4 GRU 时间信息捕捉

GRU 是一种改进的 RNN 结构，包含两个主要的门：重置门和更新门。重置门控制前一个时间步的隐藏状态对当前计算的影响，更新门决定了前一个时间步的隐藏状态和当前时间步的候选隐藏状态如何进行组合。通过门控机制，GRU 能够更好地处理长期依赖问题，相比传统 RNN 能更有效地缓解梯度消失问题。与另一种门控 RNN 变体 LSTM 相比，GRU 结构更简单、计算效率更高。

在本文中，数据经过图表示学习部分有效地学习到不同变量之间的空间依赖关系后，输入 GRU 单元进一步捕捉时间步长信息。然后，数据会通过重建模块来完成第一次重建。

为减小模型复杂度，该重建模块采用简单的编码-解码结构来实现数据的第一次重建。编码器将输入数据压缩到一个低维的潜在空间表示，可以使用一系列全连接层来结合激活函数。解码器从编码器输出的潜在表示中重建原始数据，由全连接层组成。

2.5 DDPM

由图 1 可以看出，通过图学习数据空间信息后，GRU 单元进一步捕捉时间步长信息，将更新后的数据信息再通过基于重建的模块得到第一次重建后的数据，将第一次重建后的数据再次输入扩散模型，DDPM 会从第一次重建的数据中去除噪声和异常，以获得更平滑、更接近正常数据分布的重建结果。

将第一次重建后的数据设为 $\hat{\boldsymbol{X}}^0\in\mathbb{R}^{M\times W}$，其中，$M$ 是通道变量数，W 是滑动窗口长度，其可以看作是一个单通道图像。设 $q(\hat{\boldsymbol{X}}^0)$ 为时间序列上的分布，从中可以进行抽样 $\hat{\boldsymbol{X}}^0\sim q(\hat{\boldsymbol{X}}^0)$。在正向过程中，从 $\hat{\boldsymbol{X}}^0$ 开始，在前一个输入上逐渐加入高斯噪声：

$$q(\hat{\boldsymbol{X}}^n\mid\hat{\boldsymbol{X}}^{n-1})=\mathcal{N}(\hat{\boldsymbol{X}}^n;\sqrt{1-\beta_n}\hat{\boldsymbol{X}}^{n-1},\beta_n\boldsymbol{I}) \tag{5}$$

式中：$\beta_n\in(0,1)$ 并随 n 线性增加。

在反向过程中，从损坏的时间序列开始，逐渐从输入中去除噪声。此过程建模如下：

$$p_\theta(\hat{\boldsymbol{X}}^{n-1}\mid\hat{\boldsymbol{X}}^n)=\mathcal{N}(\hat{\boldsymbol{X}}^{n-1};\mu_\theta(\hat{\boldsymbol{X}}^n,n),\tilde{\beta}_n\boldsymbol{I}) \tag{6}$$

式中：$\tilde{\beta}_n=\frac{1-\bar{\alpha}_{n-1}}{1-\bar{\alpha}_n}\beta_n$，$\alpha_n=1-\beta_n$ 并且 $\bar{\alpha}_n=\prod_{s=1}^{n}\alpha_s$。关于 $\mu_\theta(\hat{\boldsymbol{X}}^n,n)$ 的计算，是训练一个网络 ε_θ 来预测给定 $\hat{\boldsymbol{X}}^n$ 的噪声 $\varepsilon\sim\mathcal{N}(0,1)$：

$$\mu_\theta(\hat{\boldsymbol{X}}^n,n)=\frac{1}{\sqrt{\alpha_n}}\Big(\hat{\boldsymbol{X}}^n-\frac{\beta_n}{\sqrt{1-\bar{\alpha}_n}}\varepsilon_\theta(\hat{\boldsymbol{X}}^n,n)\Big) \tag{7}$$

在训练 ε_θ 网络时，需要最小化不同噪声水平 n 下的损失项：

$$L=\big\|\varepsilon-\varepsilon_\theta(\sqrt{\bar{\alpha}_n}\hat{\boldsymbol{X}}^0+\sqrt{1-\bar{\alpha}_n}\varepsilon,n)\big\|^2 \tag{8}$$

在测试时，先对输入 $\hat{\boldsymbol{X}}^0$ 添加噪声，再将其去噪 $\hat{\boldsymbol{X}}^0\xrightarrow{\text{noise}}X_{\text{noisy}}\xrightarrow{\text{denoise}}\widetilde{\boldsymbol{X}}^0$。具体来说，从 $\hat{\boldsymbol{X}}^0$ 开始，在之前的输入上加高斯噪声 M 次，直接在噪声级 M 处采样 $\hat{\boldsymbol{X}}^M\sim q(\hat{\boldsymbol{X}}^n\mid\hat{\boldsymbol{X}}^0)$，计算公式如下：

$$q(\hat{\boldsymbol{X}}^n\mid\hat{\boldsymbol{X}}^0)=\mathcal{N}(\hat{\boldsymbol{X}}^n;\sqrt{\bar{\alpha}_n}\hat{\boldsymbol{X}}^0,(1-\bar{\alpha}_n)\boldsymbol{I}) \tag{9}$$

从 $\hat{\boldsymbol{X}}^M$ 开始，迭代地对前一个输入进行 M 次去噪，最终得到 $\widetilde{\boldsymbol{X}}^0$。

2.6 评价指标

将重建的 $\widetilde{\boldsymbol{X}}^0$ 与原始样本 $\boldsymbol{X}^0$ 之间的距离作为异常分数。在给定某一阈值的情况下，若异常分数超过该阈值则认为是异常点，若低于阈值则认为是正常点。

2.6.1 PA 协议

大多数的时序数据异常检测算法会使用 PA 评估协议后再测量 F1 值。根据该协议，如果一个异

常片段由多个连续的异常点组成，只要算法正确检测到其中至少一个点，则该片段内的所有点都将被视为正确检测，并统一计为真正例(TP)。文献[7-8]已经证明该协议会严重高估算法的建模性能，无法反映真实的建模性能。为了合理地评估算法性能，文献[7]提出一个新的更加严格的 PA%k 协议。具体来说，如果一个异常片段中至少 k%的异常点被正确预测，那么该片段中的所有预测都被认为是异常的。设置 $k=0$ 相当于 PA，会导致模型性能被高估，而 $k=100$ 时，即使仅有一个点的检测错误，也会对模型施加惩罚。

2.6.2 F1@k 与 AUC@k

为了消除对 k 的依赖，计算 k 从 0 到 100 时对应的 F1 值，然后计算出这条曲线下面积，将这个评价指标称为 F1@k。

虽然 F1@k 消除了对 k 的依赖，但它仍然对给定阈值进行评估。为了消除这种依赖性，通过测量不同阈值和 k 值之间的真正例率(TPR)和假正例率(FPR)来计算受试者工作特征(ROC)曲线，ROC 曲线下的面积称为 AUC@k。因此，无论阈值如何选择，AUC@k 都可以用于比较多变量时序数据的异常检测算法。

2.7 算法具体实现

为了更详细地描述算法的流程，本文提供以下伪代码，该伪代码描述的是 GRD 的训练过程。

算法 1 GRD 算法

输入 $\boldsymbol{X}^0\in\mathbb{R}^{w\times M}$，加噪水平 N，$\lambda\in\mathbb{R}^+$，节点嵌入维度 d，Top-k//其中，w 表示窗口大小，M 表示数据维度

输出 $L=L_1+\lambda L_2$

1)根据输入得到 $\boldsymbol{X}_{\text{Embedding}}\in\mathbb{R}^{M\times d}$；

2)通过 $\boldsymbol{X}_{\text{Embedding}}$ 计算余弦相似度得到边权重 $\boldsymbol{W}_{\text{edge}}\in\mathbb{R}^{M\times M}$；

3)排序节点对之间的权重值并选择前 Top-k 得到边索引；

4)将输入数据以及边索引、边权重输入 ARMAConv 进行节点信息更新；

5)将经过节点信息更新后的数据与 $\boldsymbol{X}_{\text{Embedding}}$ 相加后输入时间信息捕捉部分，得到第一次重建后的数据 $\hat{\boldsymbol{X}}^0\in\mathbb{R}^{w\times M}$；

6)计算第一次重建的损失：$L_1=\text{MSE}(\hat{\boldsymbol{X}}^0,\boldsymbol{X}^0)$；

7)初始化 $n\sim\text{Uniform}(1,2,\cdots,N)$ 和 $\varepsilon\sim\mathcal{N}(0,\boldsymbol{I})$；

8)计算 $\hat{\boldsymbol{X}}^n$($\hat{\boldsymbol{X}}^0$ 经过 n 步加噪后得到)的扩散损失：$L_2=\left\|\varepsilon-\varepsilon_\theta(\sqrt{\bar{\alpha}_n}\hat{\boldsymbol{X}}^0+\sqrt{1-\bar{\alpha}_n}\varepsilon,n)\right\|^2$；

9)$L=L_1+\lambda L_2$；

10)反向传播以及参数更新。

3 实验

本文分别在 SMD[1]、SWaT[41]、WADI[42] 三大标准公开数据集上进行训练并检测，滑动窗口长度为 100，学习率为 0.001，训练 500 轮。

本文将 GRD 算法与基于 GNN、Transformer、VAE 的先进方法进行比较：MTAD-GAT[9] 和 GDN[4] 是基于 GNN 的先进算法，基于 Transformer 的先进算法 TranAD[43] 和 Anomaly Transformer[44] 已经在多变量时序异常检测领域表现出优秀的性能，OmniAnomaly[1] 是一种 VAE 的扩展方法，专门用于处理多变量时序数据的异常检测问题。此外，本文还评估了一个经典的异常检测算法——OCSVM[45]。同时，本文还将 GRD 算法与当前热门的基于大型预训练模型的算法——MOMENT[24] 进行比较。MOMENT 是一种专为多个时序任务设计的开源基础模型，在 Time Series Pile 数据集上进行了预训练。根据编码器的不同，MOMENT 分为 3 个版本：Small，Base 和 Large。在本次实验中，对 MOMENT-Base 的重建头模块进行微调，并在 3 个数据集上进行训练和比较。

3.1 数据集

SMD 数据集是一个多变量时序数据集，用于异常检测研究。它是从互联网公司的服务器机器上收集的监控指标数据，包括中央处理器(CPU)使用率、内存使用率、磁盘活动情况、网络流量等多个维度。SMD 数据集包含约 28 个服务器的数据，每个服务器包含 38 个监控指标。数据集中的异常主要是由于服务器的硬件或软件故障引起的，这些异常在数据集中已经被标注，便于算法的评估和比较。

SWaT 数据集是一个用于工业控制系统安全研究的数据集，特别是针对水处理系统的异常检测和入侵检测。它是从新加坡国立大学的一个物理水处理系统(SWaT 测试床)收集的数据。SWaT 测试床是一个 6 级反渗透水处理过程，模拟了真实的工业水处理环境。数据集包含正常操作期间和受到各种网络攻击时的系统传感器和执行器的读数。SWaT 数据集提供了一种评估和测试异常检测算法在工业环境中的性能的手段。

WADI 数据集是另一个用于工业控制系统安全研究的数据集，它是从一个模拟城市供水系统的物理测试床中收集的。与 SWaT 相似，WADI 测试床模拟了现实世界的供水网络，包括水库、泵站、水处理设施等。数据集包含正常运行和受到攻击情况下的系统传感器和执行器的读数。WADI 数据集

为研究工业水系统的异常检测和网络安全提供了宝贵的资源。

3 个数据的详细信息如表 1 所示。

表 1 数据集统计

Table 1 Statistics of the datasets

数据集	维度/维	训练集样本数/个	测试集样本数/个	异常率/%
SMD	38	708 405	708 420	4.16
SWaT	51	495 000	449 919	12.38
WADI	127	1 048 571	172 000	5.77

3.2 数据预处理与数据集划分

在进行时序数据异常检测之前，关键的预处理步骤包括数据清理和数据归一化。首先，数据清理涉及处理缺失值和异常值。缺失值可以通过插值、填充均值或中位数等方法来处理，以确保数据的完整性。异常值处理采用统计方法（如 z-score 或四分位距法）进行识别，随后根据具体情况替换或删除异常值，以减少其对模型的负面影响。其次，数据归一化通过标准化或归一化技术来调整数据的尺度。标准化将数据转换为零均值和单位方差的形式，适用于数据符合正态分布的情况，而归一化则将数据缩放到特定范围（通常为 0～1），适用于数据分布无明显均值和方差的情况。

原始数据集只提供了训练集和测试集，为防止信息泄露，本文严格保持时间顺序，从原始训练集的起始点开始，取前 80% 的数据作为新的训练集，取剩余的 20%数据作为验证集，测试集保持不变。

3.3 实验结果

与 GRD 算法对比的其他算法同样都是训练 500 轮，在验证集取得损失最小时保存模型参数。在训练 MOMENT-Base 时，只是微调其重建头模块而将其余部分参数冻结，根据模型开发者的建议，训练仅进行 20 轮。这种策略旨在保持预训练模型的知识，同时通过细微调整提升其在特定任务上的表现。其他的超参数（如滑动窗口长度、学习率等）和与算法相关的一些设置都按照原论文或者公开的源代码进行选择。采用 F1@k 和 AUC@k 作为评价指标进行模型评估与比较。具体的实验结果如表 2 所示，其中，F1@k 和 AUC@k 的最高分数加粗表示。

表 2 实验数据对比

Table 2 Comparison of experimental data

方法	SMD		SWaT		WADI	
	F1@k	AUC@k	F1@k	AUC@k	F1@k	AUC@k
MTAD-GAT	0.531 3	0.654 1	0.505 1	0.734 9	0.464 3	0.720 7
GDN	0.711 8	0.754 4	0.771 2	0.860 1	0.655 1	0.746 2
Diffusion	0.661 3	0.773 4	0.559 1	0.721 5	0.188 3	0.733 2
OmniAnomaly	0.680 2	0.707 1	0.530 9	0.621 7	0.441 8	0.618 1
TranAD	0.401 9	0.709 5	0.434 1	0.878 1	0.116 5	0.696 6
Anomaly Transformer	0.314 6	0.674 8	0.220 7	0.504 4	0.109 3	0.139 3
OCSVM	0.206 2	0.292 7	0.103 6	0.139 1	0.039 7	0.311 6
MOMENT-Base	0.377 6	0.540 1	0.303 2	0.480 1	0.113 6	0.181 3
GRD	**0.741 4**	**0.780 2**	**0.801 7**	**0.886 6**	**0.767 1**	**0.787 4**

在 SMD 数据集上，GRD 算法显示出最高的性能，F1@k 指标分数为 0.741 4，与 GDN 算法分数接近。这表明它在识别和重建正常行为时具有很强的能力，并有效地减少了误报和漏报。在 SWaT 数据集上，GRD 算法同样展现了优异的性能，其 F1@k 指标分数为 0.801 7，领先于其他算法。这一结果凸显了 GRD 算法在处理工业控制系统数据方面的潜力，其能够在复杂的工业环境中进行有效的异常检测。随着检测数据维度的提高，GRD 算法的优势逐渐显现。在 WADI 数据集上，GRD 算法保持了高水准的表现，F1@k 分数为 0.767 1，这显示了 GRD 算法在更广泛的工业数据集上的鲁棒性和可靠性。

4 实验结果分析

4.1 嵌入向量的降维可视化

根据前文分析，嵌入值相似的通道变量应具有较强的相关性。为验证这一假设，本文将在 SWaT 数据集上训练得到的数据嵌入向量[shape:(51,64)]使用 t-SNE[46] 方法进行降维并可视化，如图 2 所示，彩色效果见《计算机工程》官网 HTML 版，下同。

在图 2 中，最上面圈出的 4 个指标代表备用执行器数据，主要包括泵和加药泵，这些设备都是作为备用设备使用，旨在主设备发生故障或需要维

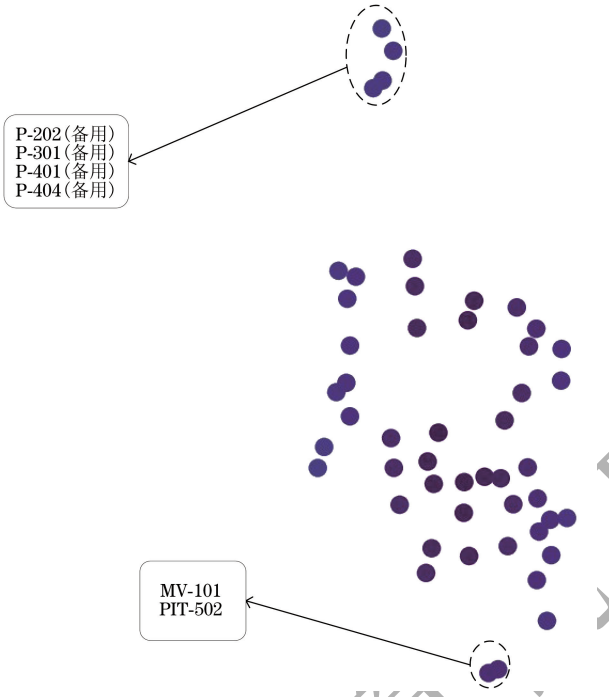

图 2 SWaT 数据集嵌入向量降维可视化

Fig. 2 Visualization of dimensionality reduction of embedded vectors in SWaT dataset

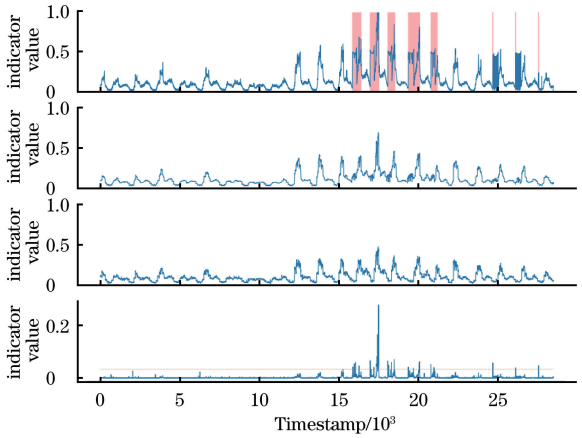

图 3 SMD 数据集 1-1 设备第一维指标

Fig. 3 The first dimension indicator of device 1-1 in the SMD dataset

护时保证水处理系统的连续运行。它们在水处理系统中执行类似的功能，解释了这些指标之间的相似性。下面圈出的两个指标分别是 MV-101(电动阀)、PIT-502(压力计)，虽然 MV-101 是一个执行设备，PIT-502 是一个监测设备，但它们的输出都是系统调整的基础。MV-101 的状态(开/关)根据系统的需要进行调整，而 PIT-502 提供的压力读数是评估系统状态并作出调整的依据。

4.2 扩散模型的作用

将包括原始数据在内的多个实验数据画在同一张图中，如图 3 所示，原始数据来自 SMD 数据集的 1-1 设备的第一维指标。

在图 3 中从上到下分别是：原始数据，第一次重建后的数据，经过扩散模型后的数据和异常分数，图中阴影部分为异常片段。GRD 算法基于经过图学习网络重建后的数据，该重建过程已对部分异常片段进行了平滑处理，基于此重建的去噪过程可以进一步平滑剩余的不规则性，经过串联两次重构，GRD 算法可以更有效地均匀剔除离群片段，使得输出的数据更加符合原始数据中正常的时序规则。

4.3 消融实验

为了验证本文算法中图表示学习以及扩散部分的必要性，逐渐排除组件来观察模型性能是如何下降的。首先，禁用图表示学习部分，仅使用扩散模型对数据进行重建。其次，为了研究扩散模型在本文算法中的重要性，仅使用第一次重建后的数据作为最终输出。最后，通过将学习到的图替换为静态完全图来分析其重要性。实验结果如表 3 所示，可以看出去除图表示部分对模型性能的影响最大。简单地将时序数据进行重建而忽略不同通道之间的联系会导致检测结果与实际情况产生较大偏差，从而验证了图表示学习部分的重要性。从去除扩散部分的实验结果中可以看出，模型在 3 个数据集上的性能皆有所下降，这是因为扩散部分能够进一步平滑时序数据中的不规则性，使得输出数据更加符合原始数据中正常的时序规则。替换学习到的图结构为静态完全图会导致 3 个数据集的性能下降，对 WADI 数据集的影响更为显著。这表明学习到的图结构对于性能提升尤其重要，特别是在大规模数据集上。

表 3 消融实验结果

Table 3 Ablation experimental results

方法	SMD		SWaT		WADI	
	F1@k	AUC@k	F1@k	AUC@k	F1@k	AUC@k
GRD	0.741 4	0.780 2	0.801 7	0.886 6	0.767 1	0.787 4
去除图表示部分	0.661 3	0.773 4	0.559 1	0.721 5	0.188 3	0.733 2
去除扩散部分	0.713 3	0.743 4	0.770 9	0.806 5	0.675 1	0.694 2
动态图结构换为静态图结构	0.707 1	0.710 8	0.741 2	0.764 3	0.533 1	0.601 1

5 结束语

本文提出的 GRD 算法针对多变量时序数据中的异常检测问题，尤其在高维场景下，通过融合 GNN 和扩散模型，有效捕捉了变量间的复杂关系和时间动态性，提高了异常检测的准确性。采用新的评估协议 PA%k 和评价指标 F1@k 更准确地反映了算法的性能，凸显了其相较于现有方法的优势。

未来研究将聚焦于进一步优化 GRD 算法，提升其可扩展性和效率，使其能够应用于更大规模和更复杂的数据集。此外，探索算法在不同领域和实际应用场景中的应用潜力，以及结合其他先进的机器学习技术来增强模型的泛化能力和鲁棒性，也是将进一步关注的方向。

参考文献

[1] SU Y, ZHAO Y J, NIU C H, et al. Robust anomaly detection for multivariate time series through stochastic recurrent neural network[C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York, USA: ACM Press, 2019: 2828-2837.

[2] 王俊, 赖会霞, 万玥, 等. 基于角度的图神经网络高维数据异常检测方法[J]. 计算机工程, 2024, 50(3): 156-165.
WANG J, LAI H X, WAN Y, et al. Angle-based graph neural network method for anomaly detection in high dimensional data[J]. Computer Engineering, 2024, 50(3): 156-165. (in Chinese)

[3] 陈何雄, 罗宇薇, 韦云凯, 等. 基于联邦学习的 SDN 异常流量协同检测技术[J]. 计算机工程, 2023, 49(3): 168-176.
CHEN H X, LUO Y W, WEI Y K, et al. Collaborative detection technology of SDN abnormal traffic based on federated learning[J]. Computer Engineering, 2023, 49(3): 168-176. (in Chinese)

[4] DENG A L, HOOI B. Graph neural network-based anomaly detection in multivariate time series[C]//Proceedings of the 35th AAAI Conference on Artificial Intelligence/33rd Conference on Innovative Applications of Artificial Intelligence/11th Symposium on Educational Advances in Artificial Intelligence. Palo Alto, USA: AAAI Press, 2021: 8-15.

[5] PARK D, HOSHI Y, KEMP C C. A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder[J]. IEEE Robotics and Automation Letters, 2018, 3(3): 1544-1551.

[6] HUNDMAN K, CONSTANTINOU V, LAPORTE C, et al. Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding[C]//Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York, USA: ACM Press, 2018: 387-395.

[7] KIM S, CHOI K, CHOI H S, et al. Towards a rigorous evaluation of time-series anomaly detection[EB/OL]. [2024-03-17]. https://arxiv.org/abs/2109.05257.

[8] SEHILI M E A, ZHANG Z H. Multivariate time series anomaly detection: fancy algorithms and flawed evaluation methodology[EB/OL]. [2024-03-17]. https://arxiv.org/abs/2308.13068.

[9] ZHAO H, WANG Y J, DUAN J Y, et al. Multivariate time-series anomaly detection via graph attention network[C]//Proceedings of the IEEE International Conference on Data Mining (ICDM). Washington D. C., USA: IEEE Press, 2020: 841-850.

[10] HO J, JAIN A, ABBEEL P. Denoising diffusion probabilistic models[J]. Advances in Neural Information Processing Systems, 2020, 33: 6840-6851.

[11] CHO K, VAN MERRIENBOER B, GULCEHRE C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation[C]//Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Stroudsburg, USA: ACL Press, 2014: 1724-1734.

[12] EMADI H S, MAZINANI S M. A novel anomaly detection algorithm using DBSCAN and SVM in wireless sensor networks[J]. Wireless Personal Communications, 2018, 98(2): 2025-2035.

[13] ZARE MOAYEDI H, MASNADI-SHIRAZI M A. ARIMA model for network traffic prediction and anomaly detection[C]//Proceedings of the International Symposium on Information Technology. Washington D. C., USA: IEEE Press, 2008: 1-6.

[14] CHEN L D, LAO K W, MA Y L, et al. Error modeling and anomaly detection of smart electricity meter using TSVD+L method[J]. IEEE Transactions on Instrumentation and Measurement, 2022, 71: 1-14.

[15] LI T Y, COMER M L, DELP E J, et al. Anomaly scoring for prediction-based anomaly detection in time series[C]//Proceedings of the IEEE Aerospace Conference. Washington D. C., USA: IEEE Press, 2020: 1-7.

[16] MUNIR M, SIDDIQUI S A, DENGEL A, et al. DeepAnT: a deep learning approach for unsupervised anomaly detection in time series[J]. IEEE Access, 2019, 7: 1991-2005.

[17] PARK J, PARK Y, KIM C I. TCAE: temporal convolutional autoencoders for time series anomaly detection[C]//Proceedings of the 13th International Conference on Ubiquitous and Future Networks (ICUFN). Washington D. C., USA: IEEE Press, 2022: 421-426.

[18] MALHOTRA P, RAMAKRISHNAN A, ANAND G, et al. LSTM-based encoder-decoder for multi-sensor anomaly detection[EB/OL]. [2024-03-17]. https://arxiv.org/abs/1607.00148.

[19] XU H W, FENG Y, CHEN J, et al. Unsupervised anomaly detection via variational auto-encoder for seasonal KPIs in Web applications[C]//Proceedings of the 2018 World Wide Web Conference. New York, USA: ACM Press, 2018: 187-196.

[20] LI Z Y, SUN Y, YANG L H, et al. Unsupervised machine anomaly detection using autoencoder and temporal convolutional network[J]. IEEE Transactions on Instrumentation and Measurement, 2022, 71: 1-13.

[21] LIANG H R, SONG L, DU J R, et al. Consistent anomaly detection and localization of multivariate time series via cross-correlation graph-based encoder-decoder GAN[J]. IEEE Transactions on Instrumentation Measurement, 2022, 71: 3139696.

[22] ZHAN J, WANG S Q, MA X D, et al. STGAT-MAD: spatial-temporal graph attention network for multivariate time series anomaly detection[C]//Proceedings of the 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Washington D. C., USA: IEEE Press, 2022: 3568-3572.

[23] DOOLEY S, KHURANA G S, MOHAPATRA C, et al. ForecastPFN: synthetically-trained zero-shot forecasting[EB/OL]. [2024-03-17]. https://arxiv.org/abs/2311.01933.

[24] GOSWAMI M, SZAFER K, CHOUDHRY A, et al. MOMENT: a family of open time-series foundation models[EB/OL]. [2024-03-17]. https://arxiv.org/abs/2402.

03885.
[25] JIN M, WANG S, MA L, et al. Time-LLM: time series forecasting by reprogramming large language models[EB/OL]. [2024-03-17]. https://arxiv.org/abs/2310.01728.
[26] CHANG C, PENG W C, CHEN T F. LLM4TS: two-stage fine-tuning for time-series forecasting with pre-trained LLMs[EB/OL]. [2024-03-17]. https://arxiv.org/abs/2308.08469.
[27] DHARIWAL P, NICHOL A. Diffusion models beat GANs on image synthesis[EB/OL]. [2024-03-17]. https://arxiv.org/abs/2105.05233.
[28] BLATTMANN A, ROMBACH R, LING H, et al. Align your latents: high-resolution video synthesis with latent diffusion models[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Washington D.C., USA: IEEE Press, 2023: 22563-22575.
[29] YANG R H, SRIVASTAVA P, MANDT S. Diffusion probabilistic modeling for video generation[J]. Entropy, 2023, 25(10): 1469.
[30] YANG D C, YU J W, WANG H L, et al. Diffsound: discrete diffusion model for text-to-sound generation[J]. ACM Transactions on Audio, Speech, and Language Processing, 2023, 31: 1720-1733.
[31] YU C H, ZHOU Q, LI J L, et al. Points-to-3D: bridging the gap between sparse points and shape-controllable text-to-3D generation[C]// Proceedings of the 31st ACM International Conference on Multimedia. New York, USA: ACM Press, 2023: 6841-6850.
[32] WYATT J, LEACH A, SCHMON S M, et al. AnoDDPM: anomaly detection with denoising diffusion probabilistic models using simplex noise[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Washington D.C., USA: IEEE Press, 2022: 19-35.
[33] ZHANG H, WANG Z, WU Z, et al. DiffusionAD: denoising diffusion for anomaly detection[EB/OL]. [2024-03-17]. https://arxiv.org/abs/2303.08730.
[34] RASUL K, SEWARD C, SCHUSTER I, et al. Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting[EB/OL]. [2024-03-17]. https://arxiv.org/abs/2101.12072.
[35] PINTILIE I, MANOLACHE A, BRAD F. Time series anomaly detection using diffusion-based models[C]// Proceedings of the IEEE International Conference on Data Mining Workshops (ICDMW). Washington D.C., USA: IEEE Press, 2023: 570-578.
[36] LUO C, LOU J G, LIN Q W, et al. Correlating events with time series for incident diagnosis[C]//Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Washington D.C., USA: IEEE Press, 2014: 1583-1592.
[37] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks[EB/OL]. [2024-03-17]. https://arxiv.org/abs/1609.02907.
[38] BIANCHI F M, GRATTAROLA D, LIVI L, et al. Graph neural networks with convolutional ARMA filters[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(7): 3496-3507.
[39] JIN W, MA Y, LIU X R, et al. Graph structure learning for robust graph neural networks[C]//Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. New York, USA: ACM Press, 2020: 66-74.
[40] SINGHAL A. Modern information retrieval: a brief overview[J]. IEEE Data Engineering Bulletin, 2001, 24(4): 35-43.
[41] GOH J, ADEPU S, JUNEJO K N, et al. A dataset to support research in the design of secure water treatment systems[M]. Berlin, Germany: Springer, 2017.
[42] MOHAMED A, OTHMAN A, GALAL W F, et al. Integrated geophysical approach of groundwater potential in Wadi Al Ranyah, Saudi Arabia, using gravity, electrical resistivity, and remote-sensing techniques[J]. Remote Sensing, 2023, 15(7): 1808.
[43] TULI S, CASALE G, JENNINGS N R. TranAD: deep Transformer networks for anomaly detection in multivariate time series data[EB/OL]. [2024-03-17]. https://arxiv.org/abs/2201.07284.
[44] XU J, WU H, WANG J, et al. Anomaly Transformer: time series anomaly detection with association discrepancy[EB/OL]. [2024-03-17]. https://transferlab.ai/pills/2022/anomaly-transformer/.
[45] SCHÖLKOPF B, PLATT J C, SHAWE-TAYLOR J, et al. Estimating the support of a high-dimensional distribution[J]. Neural Computation, 2001, 13(7): 1443-1471.
[46] VAN DER MAATEN L, HINTON G. Visualizing data using t-SNE[J]. Journal of Machine Learning Research, 2008, 9(11): 2579-2605.

编辑 陆燕菲