中国科学: 信息科学 2023 年 第 53 卷 第 5 期: 972–992
SCIENTIA SINICA Informationis
论文

基于 Transformer GAN 架构的多变量时间序列异常检测

蔡美玲[1], 汪家喜[1], 刘金平[1,2*], 唐朝晖[3], 谢永芳[3]

1. 湖南师范大学智能计算与语言信息处理湖南省重点实验室, 长沙 410081
2. 湖南师范大学计算与随机数学教育部重点实验室, 长沙 410081
3. 中南大学自动化学院, 长沙 410083
* 通信作者. E-mail: ljp@hunnu.edu.cn

收稿日期: 2022–04–01; 修回日期: 2022–06–12; 接受日期: 2022–06–23; 网络出版日期: 2023–05–12

国家自然科学基金面上项目 (批准号: 61971188)、国家自然科学基金重点项目 (批准号: 62233018)、湖南省重点领域研发计划项目 (批准号: 2016SK2017, 2019SK2161) 和湘江实验室重大项目 (批准号: 22XJ01013) 资助

摘要 基于过程中实时采集的多变量时序关联数据进行异常检测是预防工业过程事故、保障系统安全的关键环节之一. 然而, 工业多变量时间序列异常检测仍面临如下两大难题: (1) 时序数据变量间复杂的非线性关联特性缺乏有效的表达方法; (2) 正常/异常分布极度不均衡的时间序列间复杂的相关性有待深入挖掘. 本文提出一种新的基于多变量时间序列的无监督异常检测方法 —— 基于 Transformer GAN 的多变量时间序列异常检测方法 (TGAN-MTSAD). TGAN-MTSAD 采用 Transformer 网络作为生成对抗网络的基本模型, 引入了图注意力层以自动学习时序多元变量间的复杂依赖关系, 还应用了 patch 技巧使模型能够有效捕捉时间窗口内的异常细节信息, 并提出了基于重构误差与鉴别误差相结合的异常分数计算方法. 采用 3 个真实世界的数据集对所提方法进行了大量的性能验证与对比实验分析. 结果表明, TGAN-MTSAD 可以有效检测过程中的时序异常, 在大多数情况下优于基线方法, 并且具有良好的可解释性, 可用于复杂工业异常诊断.

关键词 多变量时间序列, 异常检测, Transformer, 异常分数, 图注意力

1 引言

随着物联网、人工智能等技术的发展, 传感器技术广泛应用于工业、交通、国防、科研等各个领域. 在工业自动化领域, 传感器技术是实现工业智能检测和自动控制的首要环节, 大量密集使用传感器来监控设施和系统以提高效率和安全性. 因此利用传感器采集到的数据, 通过异常检测密切监控系统的事件行为就显得尤为重要.

引用格式: 蔡美玲, 汪家喜, 刘金平, 等. 基于 Transformer GAN 架构的多变量时间序列异常检测. 中国科学: 信息科学, 2023, 53: 972–992, doi: 10.1360/SSI-2022-0133
Cai M L, Wang J X, Liu J P, et al. Transformer-GAN architecture for anomaly detection in multivariate time series (in Chinese). Sci Sin Inform, 2023, 53: 972–992, doi: 10.1360/SSI-2022-0133

异常检测是预防工业过程事故、保障系统安全的关键环节之一. 安全可靠的异常检测机制可以极大地降低系统发生故障或意外停机的风险 [1]. 现代工业控制系统通过传感器之间相关信号的多变量时间序列数据诊断异常. 由于传感器等硬件技术的发展, 可以收集到越来越可靠的时间序列数据, 其中时间序列异常检测是及时发现问题、规避风险的重要任务. 然而随着传感器数据的复杂性和维度的增加, 人类手动去监控这些数据可能会显得力不从心. 这就需要自动化的异常检测方法, 能够快速检测高维数据中的异常, 并向操作员报告这些异常, 以便能够尽快诊断异常.

因为异常检测的数据分布经常是不均衡的, 时间序列数据中正常样本远远多于异常样本. 受限于标签数据的稀缺性, 所以异常检测算法通常以无监督的方式进行训练. 多变量时间序列异常检测是一项复杂的任务, 近些年受到国内外研究学者的广泛关注. 传统的方法包括聚类方法 [2]、基于密度的方法 [3]、基于距离的方法 [4] 和基于隔离的方法 [5]. 这些传统方法通常以相对简单的方式对时间序列数据的依赖关系进行建模. 然而随着维度增加造成维度灾难, 不再允许使用传统方法来构造现实世界复杂的规则. 例如, 仅仅捕捉线性关系, 这对于许多现实世界中复杂的、高度非线性的关系是不够的.

当前, 基于深度学习的方法不仅能使高维数据集中的异常检测得以改进, 而且还能够推断时间序列之间的相关性, 因此受到了广泛的关注. 基于深度学习的方法包括 DeepAnT [6], telemanom [7], OmniAnomaly [8], LSTM-AE [9]. 最近, 生成对抗网络 (generative adversarial network, GAN) [10~14] 也展示了多变量时间序列异常检测的良好性能. 然而, 这些方法并没有很好地捕捉时间序列之间的相关性.

针对如何充分挖掘时间序列中复杂的相关性问题, 最近, Transformer 及其变体在时间序列领域已经表现出它的良好性能. 例如, 专为长序列时间序列预测设计的 Informer [15], 预测流感流行病例的时间序列模型 Deep Transformer [16] 等. 但是上述模型都是应用于时间序列预测领域的, 考虑到这是异常检测任务以及时间序列数据的特性, 在许多现实场景中仍然面临挑战: 异常往往是多样性的, 由于不同设备、不同环境、异常发生的原因各异等, 难以使模型可以学习到有效的特征表示. 通常来说, 物联网数据可能与从其他领域收集的数据相似 [17]. 也就是说物联网系统中传感器以复杂的方式高度相关, 时间序列的结构以及生成和分析数据的环境在许多方面可能会影响异常检测算法的成功.

基于此, 需要一种无监督的学习方法, 从海量的时间序列数据中学习到数据的分布同时能捕捉时间序列和传感器之间复杂的依赖关系来进行异常检测. 本文提出了基于 Transformer GAN 的多变量时间序列异常检测 (Transformer GAN-based multivariate time series anomaly detection, TGAN-MTSAD) 模型, 它引进了生成对抗思想并且利用了 Transformer 对序列数据的表示学习能力. TGAN-MTSAD 模型旨在提取多变量时间序列之间相关特征和复杂的数据分布进行重构, 同时利用 Transformer 捕捉时间序列之间复杂的依赖关系. 为了学习到更加有效的特征, 引入图注意力层来对许多具有潜在相互关系的传感器数据进行建模, 以提高模型在异常事件发生时检测和解释异常的诊断能力.

本文主要研究工作和创新之处总结如下.

(1) 提出一种基于 Transformer GAN 体系架构的无监督的时间序列异常检测方法. 据目前所知, 这是第一个使用 Transformer 网络结合 GAN 来进行时间序列异常检测的.

(2) 引入图注意力层来对传感器数据进行建模以提高该模型对异常诊断分析的能力.

(3) 引入 Wasserstein 损失函数来解决模型在训练过程中不稳定、模式坍塌等问题. 同时, 对鉴别器应用 patch 技巧使模型能够关注时间窗口中的异常细节信息. 最后, 使用基于滑动窗口的方法联合重构误差与鉴别误差来计算异常分数.

(4) 使用 3 个真实世界的工业数据集进行广泛的评估, 表明所提方法优于其他基线方法.

本文其余部分结构如下. 第 2 节整理了时间序列异常检测的相关文献以及本文工作的研究动机.

第 3 节正式阐述时间序列异常检测问题. 第 4 节详细介绍所提方法 TGAN-MTSAD 的组成部分以及基于 GAN 的模型如何进行异常检测. 第 5 节给出该方法在 3 个真实世界数据集上的实验结果. 第 6 节对本实验进行详细的分析. 最后, 对全文进行总结.

2 相关工作

本节首先回顾异常检测, 然后介绍多变量时间序列数据建模方法. 由于所提方法应用了 GAN 和图注意力层, 所以也同样总结相关工作.

2.1 异常检测

异常检测, 即发现一组数据点中和大多数数据不同的数据点. 异常检测的方法有很多, 常见的分类有基于统计的方法, 基于机器学习的方法和基于时间序列的方法等. 异常点一般都比较稀有, 即出现频率低. 所有这些方法都是基于异常点的稀有性或者与正常数据点的不一致的.

传统的方法包括聚类方法 [2]、基于密度的方法 [3]、基于距离的方法 [4] 和基于隔离的方法 [5] 等. 最近, 深度学习方法在高维数据集中的异常检测领域取得了改进, 包括自编码器 (autoencoder, AE) 及其变体. 例如, 深度自编码高斯模型 (DAGMM) [18] 联合深度自编码器和高斯混合模型为每个观测生成低维表示和重构误差. 记忆扩充自编码器 (MemAE) [19] 在自编码器中增加一个记忆模块, 并开发了一种改进的自编码器以增强重构后的异常误差. 但是上述方法并未考虑到时序特性.

本文旨在设计一种专门应用于时间序列数据的异常检测方法, 通过有效捕捉时间序列复杂依赖关系实现多变量时间序列异常检测.

2.2 多变量时间序列建模

时间序列建模的经典方法包括整合移动平均自回归 (autoregressive integrated moving average, ARIMA) 模型 [20] 和随机森林模型 [21], 然而这些方法的线性特性使得它们无法对时间序列中复杂的高维非线性特征进行建模.

为了学习非线性高维时序数据的表示和进行多变量时间序列异常检测, 基于深度学习的方法引起了研究学者的兴趣. 虽然深度学习在时间序列的应用是一个相对新的尝试, 但已经展现出惊人的前景. 深度学习方法, 例如 DeepAnT [6], telemanom [7], OmniAnomaly [8], LSTM-AE [9], ConvLSTM [22] 在实际的时间序列任务中都取得了成功. 这些方法大多基于 LSTM 神经网络. 与 Transformer 网络相比, LSTM 循环神经网络几乎无法学习跟随趋势, 而 Transformer 能够捕捉更详细的依赖关系 [23].

Transformer 网络允许数据单元之间直接连接, 能够更好地捕捉时序之间的依赖关系. 最近, 基于 Transformer 的模型 [24] 及其变体在时间序列建模方面也取得了较好的表现. 这些方法包括 Informer 模型 [15] 和 Deep Transformer 模型 [16] 等. 还有 Transformer 和其他神经网络相结合进行时间序列异常检测的方法, 例如, 基于图学习的 Transformer 异常检测模型 (GTA) [25] 利用图神经网络学习对传感器之间的关系建模然后结合时间序列数据并馈送进 Transformer 来进行异常检测. 对抗性稀疏 Transformer (AST) [26] 引入了生成对抗网络, 采用稀疏 Transformer 作为生成器来学习稀疏注意力图进行时间序列预测, 并使用附加的鉴别器来提高序列级别的预测性能.

由于 Transformer 具有良好的时间序列建模能力, 可以用于对时间序列特征的提取. 另外, GAN 和 Transformer 的结合在时间序列预测方面的优异表现, 将在时间序列异常检测中做出一些尝试.

2.3 GAN

GAN 在计算机视觉和自然语言处理领域取得了巨大的成功. 早期研究使用 GAN 来生成序列数据. 例如, C-RNN-GAN [27] 采用 GAN 架构, 使用 LSTM 神经网络作为 GAN 的基础模型来生成序列旋律数据. 最近, TimeGAN [28] 首次利用 GAN 生成时间序列. 与此同时, GAN 在异常检测领域也取得成功应用, 例如, AnoGAN [29] 通过 GAN 学习正常样本的数据分布, 然后将带有缺陷的样本映射到隐变量, 再由隐变量重构样本. GANomaly [30] 通过 3 个网络联合训练来重构样本.

此外, 一些研究使用 LSTM 神经网络结合 GAN 来进行时间序列异常检测, 例如, MAD-GAN [10]、TAnoGAN [11]、TadGAN [12]、USAD [13]、基于 GAN 的不均衡工业时间序列异常检测方法 [14].

上述研究无一例外都是使用 LSTM 作为 GAN 的基本模型的. 所以本课题将对基础模型进行创新尝试. 受图像生成领域 TransGAN [31] 的启发, 将 Transformer 和 GAN 相结合以应用于时间序列建模, 本课题参考一些研究文献 [15, 24, 26, 28, 32], 包括上述对抗性稀疏 Transformer 模型 (AST) 和 Informer 模型, 在此基础上做了调整和改进以更好地适应时间序列的特性.

2.4 图注意力网络

为了描述现实世界无处不在的关系数据, 图结构数据已经广泛应用于复杂关系的建模中. 由于图结构的强大表现力, 图神经网络 (graph neural network, GNN) 作为基于深度学习中处理图域信息的方法越来越受到重视 [33, 34]. 由于其较好的性能和可解释性, GNN 最近已成为一种广泛应用的图分析方法. 图注意力网络 (graph attention network, GAT) [35, 36] 是 GNN 中最经典的模型之一, 在传播过程中引入自注意力机制, 每个节点的隐藏状态通过注意其邻居节点来计算.

最近, 一些研究使用图注意力网络来进行时间序列异常检测. 例如, GDN [37] 首次利用图注意力网络作为特征提取器来捕捉传感器之间的关系, MTAD-GAT [38] 通过两个并行的图注意力层来捕捉多个特征和时间戳之间的关系.

为了能学习到更加有效的特征, 引入图注意力层对许多具有潜在相互关系的传感器数据进行建模, 旨在提高模型检测和解释异常的诊断能力.

3 问题描述

给定一个训练数据集 $\mathcal{X} \subseteq \mathbb{R}^{M\times k}$, 其中 k 是特征或变量的数量, M 是训练集中时间序列的长度. 测试集 $\mathcal{Y} \subseteq \mathbb{R}^{N\times k}$, 其中 N 是测试集中时间序列的长度. 异常检测的任务目标是为测试数据集分配二元标签 (其中 0 表示正常, 1 表示异常). 考虑到异常检测任务的特性, 正常数据与异常数据之间的极度不均衡性, 一般采用正常数据 (无异常数据) 进行序列数据建模, 然后基于所构建的模型对测试数据 (有异常数据) 重构以进行异常检测.

为了有效地学习 $\mathcal{X}$, 应用大小为 w 和步长为 s 的滑动窗口将多变量时间序列划分为一组子序列, 其中 $x = \{x_i | i = 1, 2, \ldots, m\}$, $x_i \subseteq \mathbb{R}^{w\times k}$, $m = \frac{M-w}{s}$ 是子序列的数量. 同样地, $z = \{z_i | i = 1, 2, \ldots, m\}$ 取自潜空间的一组子序列. 将 x 和 z 馈送至模型, 用极大极小博弈 (minimax game) 方式来训练生成器和鉴别器.

在经过足够充分的训练之后, 使用训练后的鉴别器 D 和生成器 G 来计算测试集 $\mathcal{Y}$ 中的异常分数. 在异常检测中, 测试集 $\mathcal{Y}$ 被划分为 n 个多变量子序列的滑动窗口, 其中 $y = \{y_i | i = 1, 2, \ldots, n\}$, $y_i \subseteq \mathbb{R}^{w\times k}$, $n = \frac{N-w}{s}$. 计算测试数据集中每个时间序列窗口的异常分数, 如果高于特定阈值 τ, 则判定

该窗口为异常. 给定 $\ell \in \{0,1\}^n$ 是测试数据集中的 n 维标签向量, 非零值表示检测到时间序列某个滑动窗口的异常. 例如 $\ell = \{\ell_1, \ell_2, \ldots, \ell_n\}$, 其中 $\ell_t = 1$ 或者 $\ell_t = 0$ 表示时间步 t 处的窗口为异常或正常.

4 所提方法

本节详细介绍所提出的无监督异常检测方法. 具体来说, 采用 Transformer 对时序之间的依赖关系进行建模, 同时引入图注意力层来提取不同传感器之间的关系以解释异常现象. 因为 GAN 模型在训练过程中极其不稳定, 所以引入 Wasserstein 损失函数并施加一定的梯度惩罚. 为了进一步发现系统内部潜在的异常, 联合重构误差和鉴别误差来计算异常分数.

4.1 图注意力

为了捕捉传感器之间的关系, 引入图注意力层, 在传播过程中引入自注意力机制, 每个节点的隐藏状态通过注意其邻居节点来计算. 通常来说, 给定具有 k 个节点的图, 例如, $h = \{h_1, h_2, \ldots, h_k\}$, 其中 $h_i \in \mathbb{R}^w$ 是每个节点的特征向量. 图注意力层按照式 (1) 计算每个节点的输出表示:

$$h_i' = \sigma\left(\sum_{j \in \mathcal{N}_i} \alpha_{ij} h_j\right), \tag{1}$$

其中 h_i' 表示节点 i 的输出表示, 与输入 h_i 具有相同的形状; σ 表示 Sigmoid 激活函数; α_{ij} 是衡量节点 j 对节点 i 的贡献的注意力分数, $\mathcal{N}_i$ 表示节点 i 的邻居节点; 节点 j 是节点 i 的相邻节点之一. 式 (2) 和 (3) 用于计算注意力分数 α_{ij}:

$$e_{ij} = \mathrm{LeakyReLU}(v^{\mathrm{T}} \cdot (h_i \oplus h_j)), \tag{2}$$

$$\alpha_{ij} = \frac{\exp(e_{ij})}{\sum_{r \in \mathcal{N}_i} \exp(e_{ir})}, \tag{3}$$

其中 $\oplus$ 是两个节点向量表示的拼接操作, $v \in \mathbb{R}^{2w}$ 是可学习参数的列向量, LeakyReLU 是非线性激活函数.

所提方法需要在没有先验知识的情况下检测多变量之间的相关性. 因此, 可以将多变量时间序列看作是一个有向完全图, 其中每个节点表示一个传感器的特征, 每个边表示两个对应特征之间的关系. 这样, 相邻节点之间的特征关系就可以被图注意力操作捕捉到. 图注意力操作如图 1 所示. 其中节点由 k 个顺序向量表示, 图 1 左边的彩色图形表示传感器的特征向量, 右边表示通过注意每个节点的邻居节点来计算其隐藏状态, 红色虚线表示隐藏状态的最终输出. 对具有潜在关系的传感器数据进行建模, 学习到更有效的特征的同时, 也提高了模型解释异常的能力.

4.2 Transformer

Transformer 最初在自然语言处理领域崭露头角, 大放异彩. 基于编码器 – 解码器结构的 Transformer 在时间序列异常检测领域也是一个优良的备选方案. 因为多头注意力层中的自注意力机制使得 Transformer 能够捕捉时间序列的复杂依赖关系. 编码器和解码器都包含 $\mathcal{J}$ 个完全相同的层. 每层都包含两个主要的组成部件: 多头自注意力层 (multi-head attention) 和前馈神经网络 (feedforward neural network, FNN). 多头注意力层将 $\mathcal{H} \in \mathbb{R}^{L \times d}$ (多头注意力层之前编码器或解码器的中间特征向量) 线性映射成 l 个不同的查询、键和值矩阵: $Q = \mathcal{H}W^Q$, $K = \mathcal{H}W^K$, $V = \mathcal{H}W^V$, 其中 $W^Q, W^K \in \mathbb{R}^{L \times d_k}$

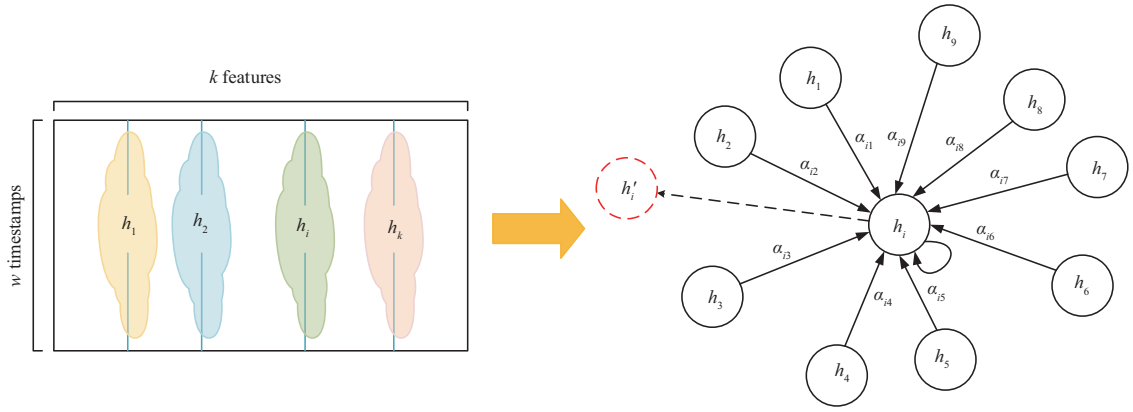

图 1 (网络版彩图) 图注意力层, 虚线表示最终的输出

Figure 1 (Color online) Graph attention layer, where the dashed circle denotes the final output

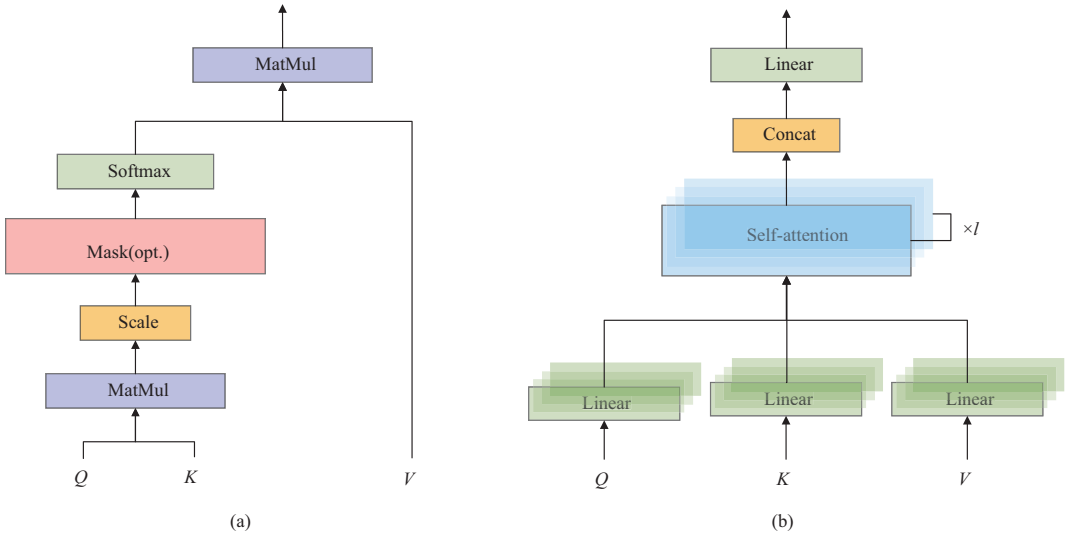

图 2 (网络版彩图) 注意力机制基本结构图. (a) 自注意力机制; (b) 多头注意力机制

Figure 2 (Color online) Fundamental structure diagram of attention mechanism. (a) Self-attention mechanism; (b) multi-head self-attention mechanism

和 $W^V \in \mathbb{R}^{L\times d_v}$ 都是可学习的参数矩阵; Q 表示查询矩阵, K 表示键矩阵, V 表示值矩阵; L 表示序列的长度, d 表示模型的维度, $d_k = \frac{d}{l}$. 自注意力机制的计算过程如图 2(a) 和式 (4) 所示.

$$\text{Attention}(Q,K,V) = \text{Softmax}\left(\frac{QK^{\mathrm{T}}}{\sqrt{d_k}}\right)V. \tag{4}$$

多头注意力层的输出 O 是 $O_1,\ldots,O_l$ 拼接后的线性映射. 多头注意力机制的计算过程如图 2(b) 和式 (5) 所示.

$$\text{MultiHead}(Q,K,V) = \text{Concat}(\text{head}_1,\ldots,\text{head}_l)W^O, \tag{5}$$

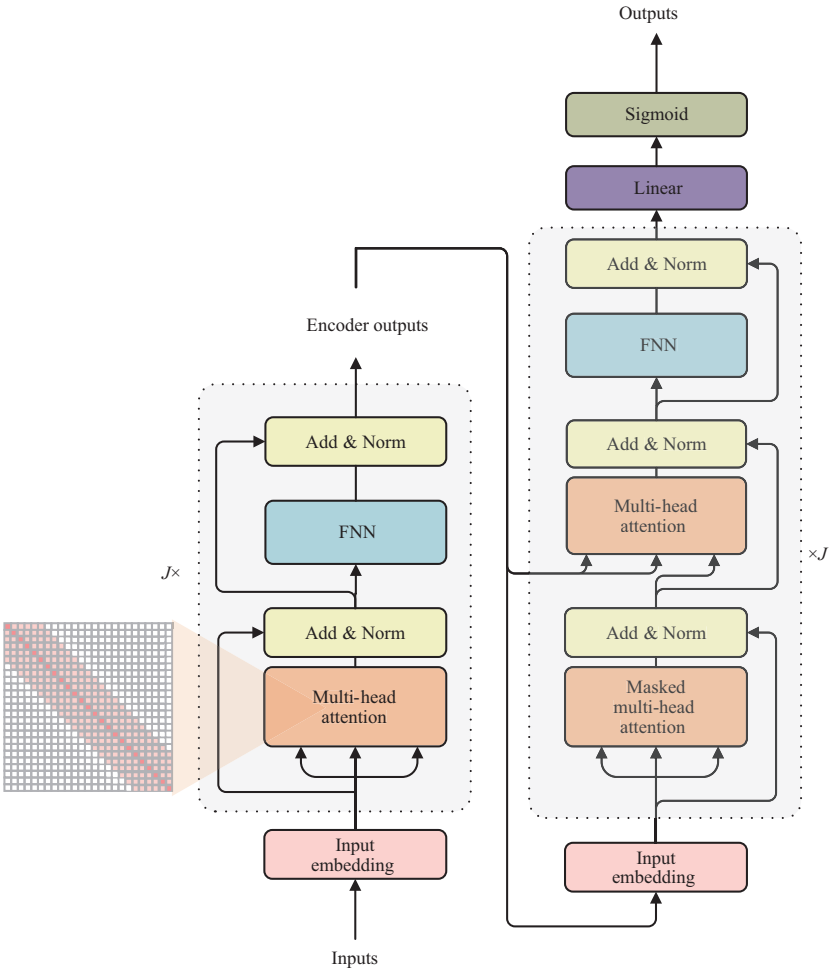

图 3 (网络版彩图) 时间序列异常检测的基本模型 Transformer

Figure 3 (Color online) The base model of the Transformer for time series anomaly detection

其中, $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$, Concat 表示多个矩阵拼接操作, O_i 表示第 i 个注意力头的输出. 前馈神经网络由两个线性层组成, 其中第一个线性层包含 ReLU 激活函数, 如下所示:

$$\text{FNN}(O) = \max(0, OW_1 + b_1)W_2 + b_2, \tag{6}$$

其中 W_1, W_2 都是可学习的权重矩阵, b_1, b_2 是偏置项. 解码器也包含多头注意力层和前馈神经网络.

本文提出一种基于 Transformer GAN 架构的时间序列异常检测模型. Transformer 模型基于多头注意力机制, 这使得它特别适合于时间序列数据: Tranformer 通过考虑其上下文 (未来 – 过去) 来同时表示每个输入序列元素, 而多个注意头可以考虑不同的表征子空间, 即输入元素之间的多个相关性, 例如, 对于时间序列, 这对应于信号的多个周期.

受图像领域 TransGAN [31] 的启发, 为了很好地将 Transformer 模型应用到时间序列上, 需要做出调整. 基础模型 Transformer 如图 3 所示. 本文的工作如下.

统一的输入表示. 在自然语言处理领域中, 原始的 Transformer 网络中通过嵌入层 (input embedding) 将每个词语映射成词向量. 考虑到时间序列数据是标量, 不需要像词语一样进行实体映

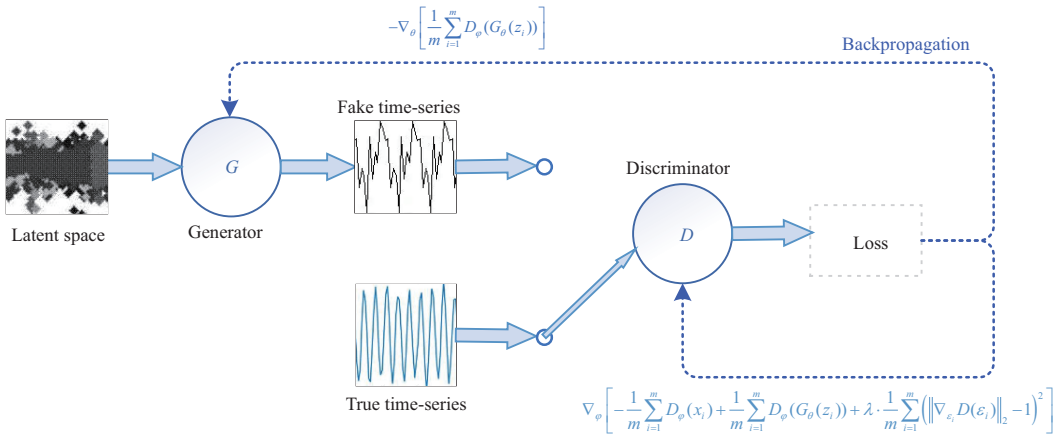

图 4 (网络版彩图) TGAN-MTSAD 模型训练过程的信息流动

Figure 4 (Color online) Information flow of the training process of TGAN-MTSAD

射. 对时间序列进行位置编码 [39] 并采用通用的可学习线性层来获得统一的输入表示.

带宽注意力. Transformer 中自注意力的一个主要效率瓶颈是成对令牌 (token) 交互所需要的时间, 其复杂度为 $\mathcal{O}(L^2)$. 为了解决这一挑战, 本工作将使用带宽注意力 (band attention) [32, 40], 在注意力图上应用一个窗口来限制其关注短期模式, 如图 3 最左侧注意力图所示. 给定一个固定的窗口大小为 c, 每个令牌关注到两边的 $\frac{1}{2}c$ 个令牌. 带宽注意力的时间和空间复杂度为 $\mathcal{O}(cL)$.

Sigmoid 层取代 Softmax 层. 文献 [41] 指出, 当使用 MSE (mean square error) 损失函数评估训练过程时, 不应该使用 Softmax 函数. 本实验正好用的就是 MSE 损失函数, 根据文献 [41] 的结论, 使用 Softmax 是不合适的, 通过查阅相关文献和进行一系列尝试, 最终选择了 Sigmoid 层, 实验结果也正好验证了所提方法的有效性.

4.3 基于 GAN 的异常检测

时间序列异常检测的基本任务是从时间序列数据中识别不正常的事件或行为. 图 4 和 5 显示所提方法的两个过程. 算法 1 给出了所提方法 TGAN-MTSAD 的伪代码. 训练过程 (图 4) 模型学习到正常时间序列数据的隐式分布. 接下来, 测试过程 (图 5) 中, 将真实的时间序列数据映射回潜空间, 并从潜空间重构序列. 重构误差将被用来计算异常分数.

为了有效地处理时间序列数据, 将 Transformer 网络作为基于 GAN 的基础模型. 生成器只使用 Transformer 网络, 在 Transformer 网络之前添加一维卷积和图注意力层作为鉴别器以更好地提取时间序列的高层语义并捕获不同传感器之间的关系.

在训练过程中, 通过对抗性训练来学习 x 的一般数据分布. 该过程同时训练生成伪时间序列数据的生成器 G 和学习区分生成的伪数据和真实数据的鉴别器 D. G 的输入是从潜空间 z 中随机选择的噪声向量. $G(z;\theta)$ 对将潜空间映射到正常数据的隐式分布空间进行建模. $D(x;\varphi)$ 函数对生成器生成数据真实概率的鉴别器进行建模. 这里, θ 和 φ 分别是生成器和鉴别器模型的参数. 该网络的损失函数使 $D(x)$ 最大化, 使 $D(G(z))$ 最小化. 经过足够多极大极小博弈的反复训练, G 和 D 将达到无法再提高的地步. 此时, G 能够生成真实的时间序列数据, 而 D 无法区分假数据和真数据.

G 和 D 都试图优化训练期间的竞争损失函数. 它们可以被认为是两个 agent 在进行一个值函数为 $V(G,D)$ 的极大极小博弈. G 试图最大化 $G(z)$ 被识别为正常的概率, 而 D 试图最小化相同的值.

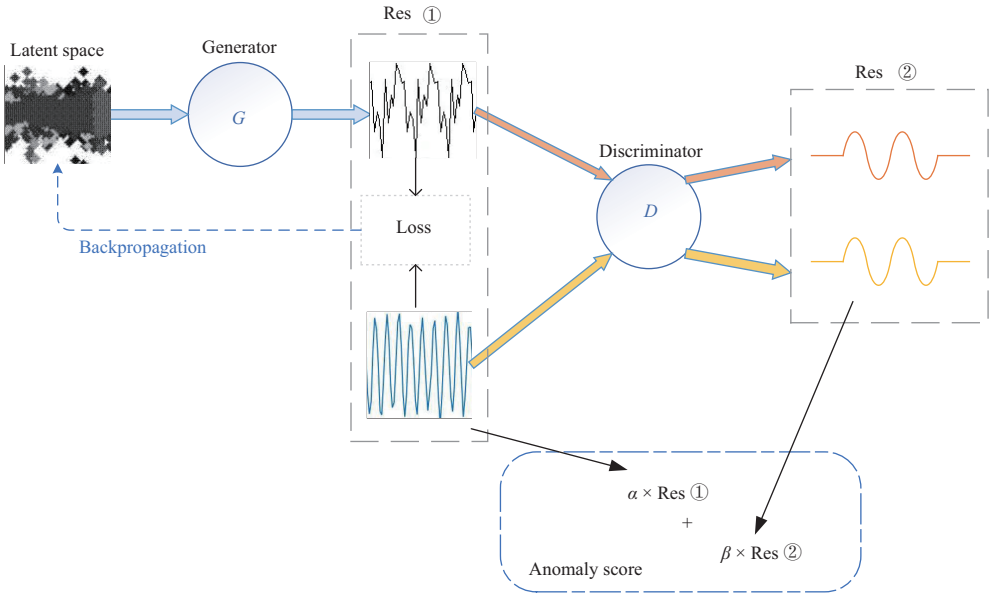

图 5 (网络版彩图) TGAN-MTSAD 模型测试过程的信息流动

Figure 5 (Color online) Information flow of the testing process of TGAN-MTSAD

值函数 $V(G,D)$ 定义为

$$\min_G \max_D V(G,D) = \mathbb{E}_{x_i\sim\mathbb{P}_{\text{data}}}[\log D(x_i)] + \mathbb{E}_{z_i\sim\mathbb{P}_z}[1-\log D(G(z_i))], \tag{7}$$

其中 $\mathbb{P}_{\text{data}}$ 是正常数据的分布, $\mathbb{P}_z$ 是潜空间的数据分布.

原始的 GAN 在训练过程中容易出现不稳定、模式坍塌等问题. 应用 GAN 时经常会发生不稳定现象, 鉴别器训练得太好会导致生成器的性能难以提上去. 文献 [42] 提出 Wasserstein 损失来优化稳定性等问题, 保证鉴别器训练得越好, 对生成器的提升就更有利. 然而 WGAN 有时也会伴随生成样本质量低、难以收敛等问题. 由于 Wasserstein 损失为了保证 Lipschitz 限制, 采用权重截断的方法, 然而这样的方法过于简单粗暴. 文献 [43] 提出给鉴别器损失施加梯度惩罚 (gradient penalty, GP) 来避免模型建模能力弱化等问题. 改进的目标函数为

$$\min_G \max_{D\in\mathcal{D}} V(G,D) = \mathbb{E}_{z_i\sim\mathbb{P}_z}[D(G(z_i))] - \mathbb{E}_{x_i\sim\mathbb{P}_{\text{data}}}[D(x_i)] + \lambda\cdot\mathbb{E}_{\varepsilon_i\sim\mathbb{P}_\varepsilon}[(\|\bigtriangledown_{\varepsilon_i}D(\varepsilon_i)\|-1)^2], \tag{8}$$

其中 $\mathcal{D}$ 是 1-Lipschitz 函数集.

为了进一步挖掘潜在异常, 将联合重构误差和鉴别误差来计算异常分数. 在实践中, 异常观测通常形成连续的分段, 因为它们是以连续的方式发生的 [8,38]. 正如文献 [44] 中所讨论的, 对于时间序列异常检测任务, 人们通常不关心点异常. 实际的情况是连续的异常段内任何点被归类为异常, 即一旦检测到一个窗口内所包含的一个点是异常的, 则该窗口就被标记为异常. 基于此, 将时间序列数据划分为多个子序列, 使用滑动窗口的方式来进行异常检测. 但是, 鉴别器的输出通常是一个判断生成器生成或真实数据表示概率的数, 并不能关注到时间窗口当中的异常细节信息. 受 PatchGAN [45] 的启发, 鉴别器的输出可以不是标量. 将 patch 技巧应用基于 GAN 的模型不仅能够提高模型捕捉时间窗口当中的异常细节信息的能力, 还能提高异常检测的各项指标.

从重构误差和鉴别误差两个方面来定义异常分数, 如式 (9) 所示.

Algorithm 1 Time series anomaly detection algorithm used in the TGAN-MTSAD method

Input: training data x, testing data y;

At training model stage:
  Initialize generator G, discriminator D;
  while θ has not converged **do**
    for $t = 1, \ldots, n_{\text{epoch}}$ **do**
      Sample a random number $\mu \sim U(0,1)$;
      $\varepsilon_i = \mu x_i + (1-\mu)G(z_i)$;
      Update parameters of D according to the gradient:
        $\varphi \leftarrow \nabla_\varphi[-\frac{1}{m}\sum_{i=1}^{m} D_\varphi(x_i) + \frac{1}{m}\sum_{i=1}^{m} D_\varphi(G_\theta(z_i)) + \lambda \cdot \frac{1}{m}\sum_{i=1}^{m}(||\nabla_{\varepsilon_i} D(\varepsilon_i)||_2 - 1)^2]$;
    end for
    Sample z_i from latent space z;
    Update parameters of G according to the gradient:
      $\theta \leftarrow -\nabla_\theta[\frac{1}{m}\sum_{i=1}^{m} D_\varphi(G_\theta(z_i))]$;
  end while
Mapping test data to the latent space:
  Initialize latent space z;
  while ξ has not converged **do**
    Sample z_i from latent space z;
    Update parameters of z according to the gradient:
      $\xi \leftarrow \nabla_\xi[\frac{1}{n}\sum_{i=1}^{n} \|y_i - G(z_i)\|]$;
  end while
  Record the optimal latent space z' according to the corresponding y;
At anomaly detection stage:
  Calculate Res ①: $\|y_i - G(z_i')\|$;
  Calculate Res ②: $\|D(y_i) - D(G(z_i'))\|$;
  Calculate anomaly score $A(y_i) = \alpha\cdot$Res ①$+ \beta\cdot$Res ②;
  if $A(y_i) \geqslant \tau$:
    $\ell_i \leftarrow 1$;
  else:
    $\ell_i \leftarrow 0$;
  end if

Output: a set of binary labels ℓ.

重构误差 重构异常信息的过程中会丢失信息, 异常与正常信息的分布截然不同, 因此可以使用测试数据和重构数据之间的误差来识别异常.

鉴别误差 训练好的鉴别器可以区分真实数据和异常数据, 它可以作为异常检测的直接工具. 为了进一步挖掘潜在的异常, 将联合重构误差来计算异常分数:

$$A(y_i) = \alpha \cdot \|y_i - G(z_i')\| + \beta \cdot \|D(y_i) - D(G(z_i'))\|, \tag{9}$$

其中 $\alpha + \beta = 1$, $\|\cdot\|$ 表示 L2 范数.

当新的时间窗口到达时, 使用该模型生成异常分数. 如果窗口的异常分数高于预定阈值, 判定新到来的时间窗口为异常. 由于可以采用不同的方法 (如极值理论 [46]) 来设置阈值, 因此相同的异常检测模型可能会在不同阈值下导致不同的预测性能. 本实验在所有可能的阈值上应用网格搜索以搜索最佳结果.

表 1 数据集的统计数据

Table 1 The statistics of datasets

	SWaT	WADI	SVS
Number of sensor	25	67	88
Length of training data	99360	241921	10362
Length of testing data	89984	15701	7057
Anomaly ratio (%)	11.99	7.09	12.33

5 实验验证

5.1 数据集和评价指标

为了证明所提方法的有效性, 在 3 个真实世界的异常检测数据集上进行大量的实验性能验证.

SWaT. 安全水处理 (the secure water treatment, SWaT) 系统[1)] 是水处理的操作试验台, 代表了大城市大型现代化水处理厂的小规模版本. 最初于 2015 年 5 月开展网络攻击调查研究. 该试验台是一个生产过滤水的缩小的水处理厂. 这些收集到的原始数据在 11 天内每秒采样一个观测点.

WADI. 配水 (the water distribution, WADI) 系统[2)] 是 SWaT 的延伸, 它同样配备了化学加药系统、增压泵和阀门、仪器和分析仪. 配水系统有 3 个控制过程. 该试验台代表真实城市供水系统的缩小版本. 这些数据在 15 天内每秒收集一次.

SVS. 蒸排系统 (stream ventilation system, SVS) 是炼钢厂的轴承数据集, 该数据集从炼钢厂某个机组的五台风机设备上收集. 在炼钢厂的主机区设备中, 蒸排装置是重要的辅助设备, 其主要功能是将主机区冷却铸铁所产生的大量蒸汽排出厂房之外, 从而为机组提供安全干净的工作环境. 在机组不停机的情况下, 使用 L2 温度传感器、3836 振动测试分析仪每间隔一段时间收集一次数据.

上述 3 个数据集的统计数据如表 1 所示. 为了与其他方法进行客观的比较, SWaT 和 WADI 的原始数据样本下采样为每 5 秒一次并选取中位数. 3 个数据集都由该领域专家标记, 异常数据只存在于测试数据中.

本文使用标准的评价指标, 即准确率 (P), 召回率 (R) 和 $F1$ 分数来评估所提方法的异常检测性能:

$$P = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}, \tag{10}$$

$$R = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{11}$$

$$F1 = \frac{2PR}{P + R}, \tag{12}$$

其中 TP 表示真阳性的数量, 即异常样本被检测为异常; FP 表示假阳性的数量, 即正常样本被检测为异常; FN 表示假阴性的数量, 即异常样本被检测为正常.

5.2 数据预处理和实验设置

在训练前进行数据标准化, 以提高模型的稳健性. 所以这里将传感器测量值进行最大最小归一化

1) https://itrust.sutd.edu.sg/itrust-labs-home/itrust-labs_swat/.
2) https://itrust.sutd.edu.sg/itrust-labs-home/itrust-labs_wadi/.

表 2 Transformer 网络组件详解

Table 2 The Transformer network components in details

Component	Details
Encoder (J=3)	
Inputs	Embedding ($d = 64$)
Band self-attention block	Multi-head band attention ($l = 8$, $c = 24$)
	Add, LayerNorm, Dropout ($p = 0.2$)
	FNN ($d_{\text{inner}} = 128$), ReLU
	Add, LayerNorm, Dropout ($p = 0.2$)
Decoder (J=3)	
Inputs	Embedding ($d = 64$)
Masked BSB	Add mask on attention block
Band self-attention block	Multi-head band attention ($l = 8$, $c = 24$)
	Add, LayerNorm, Dropout ($p = 0.2$)
	FNN ($d_{\text{inner}} = 128$), ReLU
	Add, LayerNorm, Dropout ($p = 0.2$)
Final	
Outputs	FC ($d = d_{\text{out}}$), Sigmoid

处理, 如式 (13) 所示, 数据预处理将应用于训练集和测试集:

$$\hat{\mathcal{X}} = \frac{\mathcal{X} - \min \mathcal{X}}{\max \mathcal{X} - \min \mathcal{X}}, \tag{13}$$

其中 $\max \mathcal{X}$ 和 $\min \mathcal{X}$ 分别是训练集的最大值和最小值.

本实验使用 CUDA 10.2 的 PyTorch 1.8.1 版本来实现该方法. 在 NVIDIA GeForce RTX 2060 GPU 上进行了所有实验. 对于时间序列异常检测任务, 将滑动窗口大小设置为 30, 步幅大小设置为 5. 通用模型输入嵌入维度设置为 64. 对于多头局部自注意力机制, 头的数目设置为 8, 总共有 3 个编码器层和 3 个解码器层, 全连接层的维度设置为 64. 此外, 还应用 Dropout 策略, Dropout 率设置为 0.2 以防止过拟合. 使用 Adam 优化器对模型进行训练, 学习速率初始为 1E−4, 优化器的超参数 β_1 和 β_2 分别为 0, 0.9, 梯度惩罚因子 λ 设为 10, 对该模型进行 100 个周期的训练. 表 2 总结了 Transformer 网络组件的细节. 对于带宽注意力机制, 这里令 $l = 8, c = 24$, 并且加入残差连接 (Add), 前馈神经网络 (内层的维度 $d_{\text{inner}} = 128$) 和一个 Dropout 层 ($p = 0.2$). 异常分数通过重构损失和鉴别损失的加权和计算, 根据经验选择加权权重值多次实验以产生最佳结果.

5.3 与基线方法进行比较

现有的时间序列异常检测方法大致分为两类: 传统方法和基于深度学习的方法. 将所提方法与传统方法和基于深度学习的 SOTAs 作为基线方法进行比较.

在实际应用中, 在某个阈值上取得优异的 $F1$ 分数比在大多数阈值上取得一般的表现更为重要. 因此, 按照文献 [8] 的评价方法, 对于数据集上特定方法的性能, 列举了所有可能的异常阈值, 通过网格搜索找到最佳 $F1$ 分数作为主要评价指标. 表 3 [47~49] 列出所有基准数据集上每种方法的最佳 $F1$ 分数及其相应的准确率和召回率.

表 3　时间序列异常检测中不同基线方法的性能比较

Table 3　Performance comparison of time-series anomaly detection baselines

Method	SWaT			WADI			SVS		
	P (%)	R (%)	$F1$	P (%)	R (%)	$F1$	P (%)	R (%)	$F1$
Traditional methods									
PCA [47]	24.92	21.63	0.2316	33.53	5.63	0.0964	28.33	41.61	0.3371
KNN [4]	7.83	7.83	0.0783	7.76	7.75	0.0775	43.15	14.48	0.2168
FB [48]	10.17	10.17	0.1017	8.60	8.60	0.0860	26.77	36.90	0.3103
IF [5]	95.12	58.84	0.7271	29.92	15.83	0.2071	8.76	31.84	0.1374
Deep learning methods									
DAGMM [18]	27.46	69.52	0.3937	54.44	26.99	0.3609	42.32	51.22	0.4635
LSTM-VAE [49]	96.24	59.91	0.7385	87.79	14.45	0.2482	71.26	33.25	0.4559
OmniAnomaly [8]	97.91	75.76	0.8542	84.66	89.32	**0.8693**	88.12	51.20	0.6477
MAD-GAN [10]	98.97	63.74	0.7754	41.44	33.92	0.3730	45.63	75.12	0.5677
USAD [13]	98.70	74.02	0.8460	64.51	32.20	0.4296	57.12	42.13	0.4849
GDN [37]	**99.35**	68.12	0.8082	**97.50**	40.19	0.5692	**89.33**	12.36	0.2172
MTAD-GAT [38]	93.42	67.74	0.7853	78.96	85.33	0.8202	36.51	48.73	0.4174
TGAN-MTSAD	97.54	**83.81**	**0.9016**	76.95	**98.32**	0.8633	71.29	**85.74**	**0.7785**

结果表明, 4 种传统异常检测方法 (PCA, KNN, FB, IF) 在高维数据中全部失效, 无法进行多变量时间序列异常检测. 由于现实世界数据的复杂性和维度灾难, 传统方法难以对时序之间复杂的依赖关系和高维数据进行建模. 基于深度学习的方法可以从高维度的数据中自动提取有用的特征, 如表 3 所示, 它们在 3 个数据集上都具有更好的表现. 与第二好的方法相比, TGAN-MTSAD 在 SWaT 数据集上的 $F1$ 分数仍然提高了 5.55%, 在 SVS 数据集上提高了 20.19%. WADI 数据集比 SWaT 具有更强的不均衡性, 并且具有更高的维度. 因此, 即使在数据不均衡和高维的攻击场景中, 该方法也表现出显著的有效性, 这对现实应用具有重要价值. 这些结果清楚地证明了 TGAN-MTSAD 能够有效地进行时间序列异常检测.

6　结果分析

本节通过一系列综合实验来分析图注意力层和 Transformer 网络的有效性, 并对此进行更深入的详细讨论. 另外, 应用 Wasserstein 损失以及 patch 技巧进行定性实验, 然后进一步探究 patch 技巧对基于 GAN 模型生成器和鉴别器的影响, 研究 patch 技巧对实验性能的提升. 最后, 对窗口大小、潜空间的维度进行了灵敏度研究, 分析这两个参数如何影响所提方法的性能.

6.1　图注意力的有效性

炼钢厂的双支撑单吸式风机结构, 如图 6 所示. 不同风机安装传感器的个数和位置不同. 风机传感器振动检测点 1, 2, 3, 4 分别表示电机自由侧 (motorfree)、电机负荷侧 (motorload)、风机负荷侧 (fanload) 和风机自由侧 (fanfree). 从炼钢厂获取到的数据主要有各测点原始数据、冲击平均值、速度有效值、轴承温度等.

如图 7 所示, 在正常过程 (绿色片段) 中, 风机负荷侧温度 (Fanload_Temper) 和电机负荷侧温度

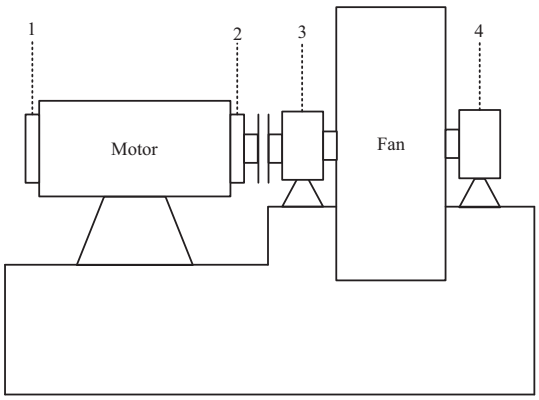

图 6 蒸排风机测点示意图

Figure 6 Schematic diagram of measurement points on the steam ventilator

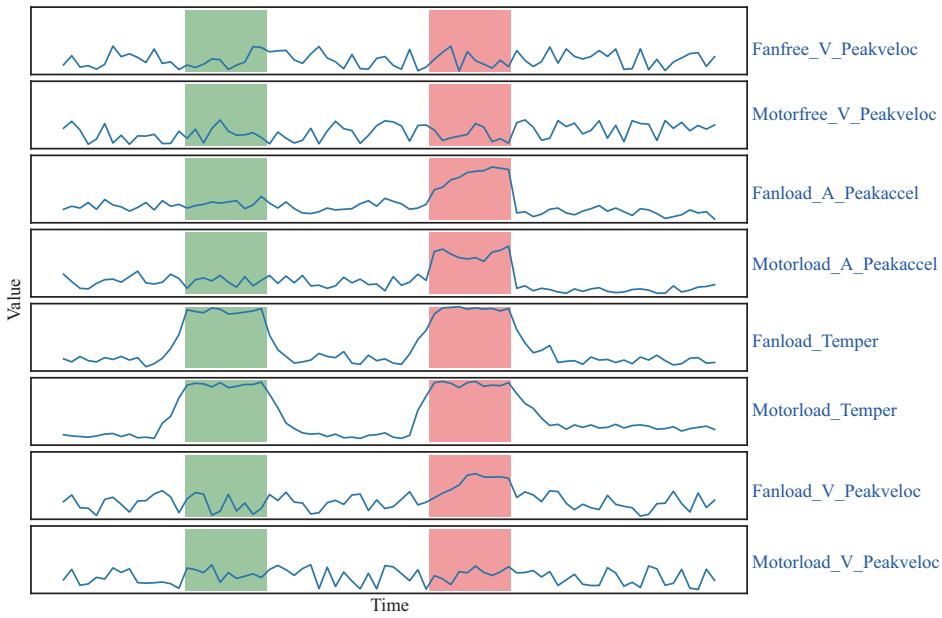

图 7 (网络版彩图) 多变量时间序列输入的案例

Figure 7 (Color online) An example of multivariate time-series inputs

(Motorload_Temper) 明显激增, 但系统仍然处于健康状态, 这两个特征有一致的趋势. 然而, 在异常过程 (红色片段) 中, 电机负荷侧温度显示出与其他速度指标也有一致的趋势, 表明风机轴承的高频振动同时也产生大量热量. 因此, 在多变量时间序列异常检测系统中, 必须考虑到不同时间序列之间的关联性.

异常诊断的能力很大程度上是利用图注意力层. 在图 8 中, 可视化基于式 (2) 和 (3) 图注意力层计算出的注意力分数. 在正常情况下, 注意力分数在图 8(a) 可见. 可以观察到, 该模型能够准确地学习到与电机负荷侧温度最相关的特征. 如图 8(b) 所示, 蒸排系统中, 电机负荷侧温度与风机负荷侧温度之间的关联性, 相较其他特性间的关联性要强得多.

图 8(a) 和 (b) 分别对应一个失败案例和一个成功案例, 通过这两个案例来分析该模型的优势. 首

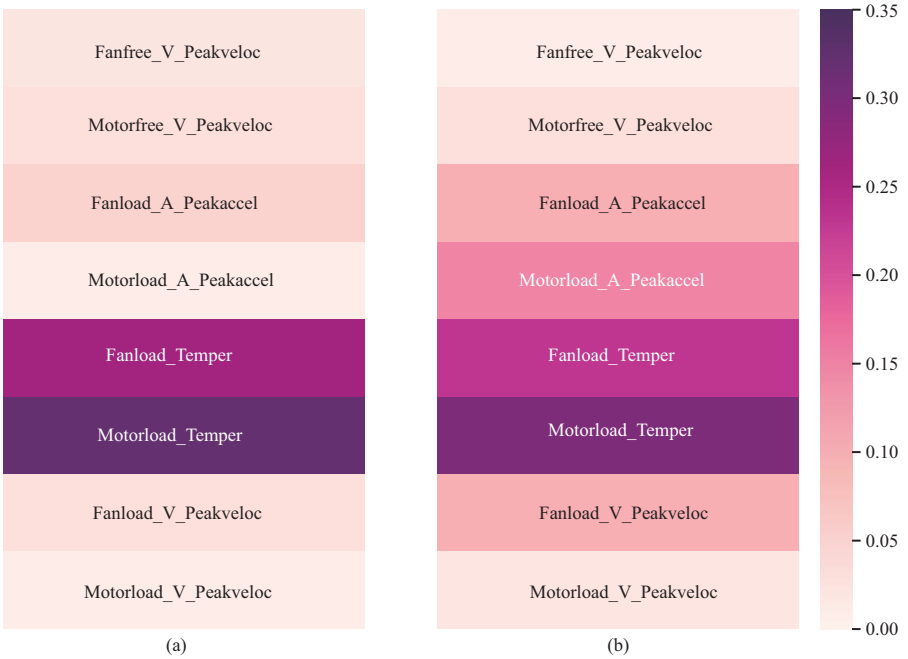

图 8　(网络版彩图) 关于图 7 电机负荷侧温度的注意力分数的说明

Figure 8　(Color online) Illustration of attention scores for the temperature of motor load side in the case of Figure 7. (a) False positive; (b) true positive

先, 本文分析图 8(a) 中的一个假阳性案例 (对应图 7 的绿色片段), 表示一个正常的案例, 被模型错误地检测为异常. 经过详细检查, 设备安装时, 电机侧半联轴器和风机侧半联轴器距离过远, 导致两个半联轴器连接后, 引入过大的轴心力, 在轴心力作用下, 两个轴承内圈向联轴器方向偏移, 造成轴承工作间隙偏小和偏载, 于是增大了滚动体的摩擦阻力, 使其运转时产生过多热量. 但轴承工作间隙偏小也会使设备振动值下降, 所以设备运转时各项振动值指标仍然达到正常. 然而温度突然骤升, 异常检测算法很容易将其作为异常情况处理. 工作人员现场排除故障后, 将半轴联器移动到合适的位置, 各项指标恢复正常标准, 系统也恢复正常运行. 该方法虽然没有很好地检测出异常段, 但提供了有用的见解, 在快速缩小异常排查范围并最终确定以及定位故障原因方面也发挥着重要作用.

接下来, 在图 8(b) 中, 展示了该模型的一个真阳性案例 (对应图 7 的红色片段). 该方法能充分考虑到特征关系, 可以通过这些特征之间的相关性来推断异常可能发生的原因. 在这个案例中, 风机轴在长期使用时受到一定程度的变形和磨损, 导致风机负荷侧轴承滚动体偏离原设计轨道高速旋转, 于是滚动体与内外圈摩擦产生大量热量, 也造成了轴承温度骤升. 与此同时, 在运行时设备各项振动值指标也飙升. 现场排除故障后, 更换了风机负荷侧轴承, 系统恢复正常运行. 该方法在处理这些情况时很有优势, 从而在很大程度上减少了监测系统的错误漏报数量.

6.2　Transformer 的有效性

为了研究该方法使用 Transformer 作为基本模型的有效性, 将一些经典序列模型, 如 RNN, GRU, LSTM 作为生成器和鉴别器代替 Transformer, 与该模型形成参照对比. 同时, 鉴于 GAN 模型中大多使用 CNN 作为基础模型. 考虑到时间序列的特性, 这里使用一维卷积神经网络 (1D-Conv) 作为基础模型进行参照对比. 另外, 为了排除 Transformer 中注意力机制的影响, 将带有注意力机制的 LSTM

表 4 不同基本模型的性能比较

Table 4 Performance comparison of different base models

Base model	SWaT			WADI			SVS		
	P (%)	R (%)	$F1$	P (%)	R (%)	$F1$	P (%)	R (%)	$F1$
Transformer	**97.54**	83.81	**0.9016**	76.95	**98.32**	**0.8633**	71.29	**85.74**	**0.7785**
RNN	75.63	62.32	0.6833	65.26	73.18	0.6899	61.20	74.12	0.6704
GRU	65.88	72.13	0.6886	86.35	53.84	0.6633	73.25	45.20	0.5590
LSTM	93.51	76.11	0.8392	**90.21**	77.12	0.8315	**78.79**	63.10	0.7008
LSTM-Attn	88.36	**85.72**	0.8702	56.32	88.69	0.6889	45.12	89.63	0.6002
1D-Conv	89.25	63.22	0.7401	61.43	72.54	0.6652	72.33	72.58	0.7245

图 9 (网络版彩图) Transformer 与 LSTM 作为生成器的重构

Figure 9 (Color online) Reconstruction of Transformer vs. LSTM as the generator

(LSTM-Attn) 也作为基本模型进行参照对比. 从表 4 中可以看出, 该方法的 $F1$ 分数在 SWaT, WADI 和 SVS 3 个数据集上分别高于第二好的方法 3.60%, 3.82% 和 7.45%.

为了进一步探索 Transformer 网络作为基本模型的特征提取能力, 将使用 Transformer 和 LSTM 网络作为生成器来重构电机负荷侧温度突然飙升 (对应图 7 的绿色片段) 的这段时间. 从图 9 中可以看出, 蓝色线表示真实电机负荷侧温度飙升的示例, 橙色线表示基于 Transformer 网络重构的, 绿色线表示 LSTM 网络重构的. Transformer 能够捕捉更详细的依赖关系并对其进行重构, 而 LSTM 几乎没有学习到任何跟随趋势.

通过对比实验证明 Transformer 强大的时序特征提取能力. 一种可行的解释是, 递归学习机制核心属性限制了建模过程的顺序性. 过去信息必须通过过去的隐藏状态来保留, 这限制了模型的长期序列建模能力. 另外, 在计算机视觉领域中, 使用 CNN 是为了有效获取图像像素中的局部强关联性特征. 考虑到时间序列数据存在长跨度的关联性, CNN (1D-Conv) 难以捕捉到时间序列数据之间复杂的依赖关系, 进而影响多变量时间序列异常检测的性能. 使用 Transformer 不仅可以关注局部信息, 还可以构建全局关键点之间的联系, 学习到时序之间复杂的依赖关系. 选用 Transformer 网络作为基于 GAN 的异常检测模型的基础模型, 能够学习到时间序列数据的隐式分布, 有效提高异常检测模型的性能. Transformer 采用非顺序学习方式, 强大的自注意力机制使时间序列中任意表征之间的上下文距离缩小为 1, 这对序列建模具有重要意义.

表 5　使用 Wasserstein 损失函数和 patch 技巧的性能比较

Table 5　Performance comparison of the Wasserstein loss and patch trick

Wasserstein	patch	SWaT			WADI			SVS		
		P (%)	R (%)	$F1$	P (%)	R (%)	$F1$	P (%)	R (%)	$F1$
✓	✓	**97.54**	**83.81**	**0.9016**	**76.95**	**98.32**	**0.8633**	**71.29**	**85.74**	**0.7785**
✓	×	78.88	63.29	0.7023	68.22	70.40	0.6929	63.25	36.32	0.4614
×	✓	89.23	81.29	0.8508	59.25	71.72	0.6489	45.98	12.44	0.1958
×	×	78.92	42.30	0.5508	66.58	89.23	0.7626	69.12	23.15	0.3468

表 6　未使用 (左边) 和使用 (右边) patch 技巧的重构损失 MSE

Table 6　Reconstruction loss MSE without (left) and with (right) the patch trick

Length of window	SWaT		WADI		SVS	
	Without patch	With patch	Without patch	With patch	Without patch	With patch
5	0.09	**0.07**	**0.28**	0.39	0.24	**0.07**
10	0.22	**0.13**	0.17	**0.08**	0.19	**0.13**
20	**0.09**	0.16	0.20	**0.12**	0.26	**0.18**
30	0.12	**0.10**	0.17	**0.13**	0.35	**0.22**
50	0.13	**0.07**	0.16	**0.08**	0.32	**0.12**
100	0.43	**0.19**	0.71	**0.46**	0.37	**0.33**

6.3　patch 技巧的有效性

通过逐次禁用 Wasserstein 损失或 patch 技巧来检查其对模型的影响, 将该方法与定性实验的简化方法进行比较来展示使用 Wasserstein 损失和 patch 技巧的有效性. 表 5 中的定性结果表明, 相较于未使用 Wasserstein 损失或 patch 技巧的简化方法, 该方法的所有评价指标在 3 个数据集上都远好于这些简化方法. 它实现了最好的性能, 显著提升了模型的各项指标.

在 GAN 训练过程中, 可能伴随生成样本质量低、训练不稳定的问题, 通过引入 Wasserstein 损失函数, 能够改善生成样本的质量. 另外, 为了进一步探究 patch 技巧对基于 GAN 模型的生成器和鉴别器的影响, 首先对训练集中的正常样本进行采样重构以探究 patch 技巧对生成器重构样本的影响. 这里与未使用 patch 技巧的 GAN 模型进行对比, 应用在不同窗口大小上, 使用 MSE 来评估生成器重构能力. 在表 6 中, 左边是未使用 patch 技巧生成器重构样本的 MSE, 右边是使用 patch 技巧生成器重构样本的 MSE. 在 3 个数据集上使用 patch 技巧的生成器重构样本能力要优于未使用 patch 技巧的生成器.

图 10 中的案例取自于电机负荷侧温度 (对应图 7 的红色片段) 突然飙升这段时间, 这里将使用一段正常与异常发生之间的滑动窗口 (窗口大小为 20), 以探究 patch 技巧对鉴别器的影响. 经过训练后的生成器, 可以被当成一个正常数据分布的隐式模型. 鉴别器的输出通常是一个判断真实数据表示概率的数, 并不能关注到时间窗口当中的异常细节信息. 如图 10 所示, 第 1 行表示真实数据, 真实数据中绿色表示正常, 红色表示异常; 第 2 行表示将真实数据映射回潜空间, 然后馈送生成器和鉴别器的 patch 输出; 第 3 行表示将真实数据馈送鉴别器的 patch 输出. 这里如果不使用 patch 技巧, 整个滑动窗口的输出分别是 0.47 和 0.32, 通过观察, 使用 patch 技巧能够显著提升鉴别器捕捉时间窗口当中的异常细节信息的能力.

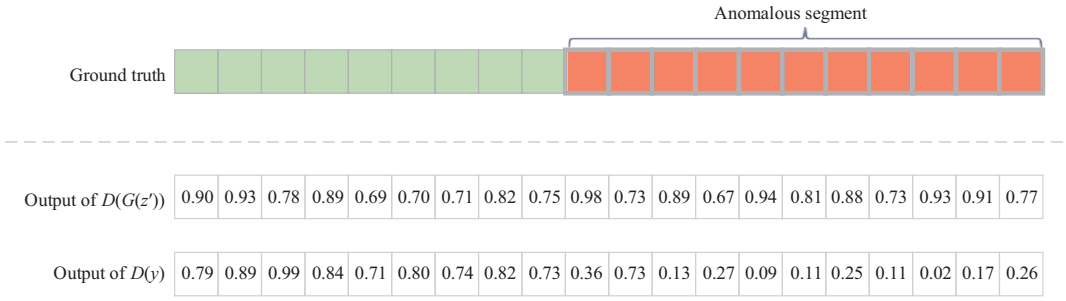

图 10　(网络版彩图) 使用 patch 技巧的鉴别器输出

Figure 10　(Color online) Discriminator's outputs using the patch trick

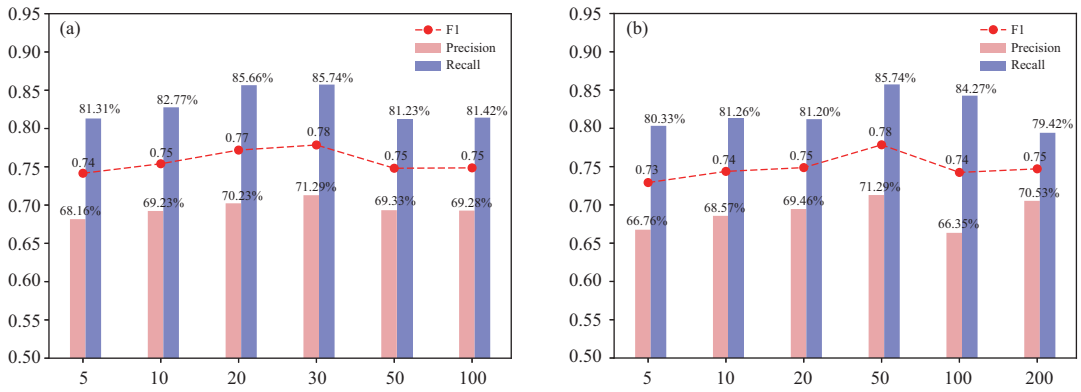

图 11　(网络版彩图) 不同参数的影响. (a) 窗口大小; (b) 潜空间维度

Figure 11　(Color online) Effect of different parameters. (a) Window size; (b) dimension of the latent space

6.4　参数分析

本小节研究两个参数对该方法性能的影响. 所有实验都在 SVS 数据集上进行.

首先, 研究的因素是数据中不同窗口大小对异常行为的影响. 因为该模型是基于滑动窗口的方式来进行时间序列异常检测的, 窗口大小对检测到的异常行为有直接影响. 图 11(a) 显示 6 种不同窗口大小 $w \in [5, 10, 20, 30, 50, 100]$ 的检测结果. 当窗口大小 $w = 30$ 时, 检测效果最好. 如果窗口太大, 异常隐匿在其中, 这使得它不容易被模型检测到; 然而, 如果窗口太小, 模型无法获取到上下文信息, 同样也很难进行推理. 所以选择一个合适大小的窗口尤为重要.

其次, 研究的因素是潜空间维度对模型性能的影响. 潜空间 z 位于一个 d_z 维空间中. 图 11(b) 显示了 $d_z \in [5, 10, 20, 50, 100, 200]$ 的检测结果. 结果显示, 将真实数据映射到一个非常小的 d_z 维度的潜空间会造成大量的信息损失, 从而无法恢复, 最终影响模型的性能; 另一方面, 使用大的 d_z 值会造成训练数据在网络中的存储, 从而导致性能下降. 因此, d_z 的中间值对性能没有很大影响, 显示出相对较高且稳定的 $F1$ 分数.

7　结论

本文提出了一种基于 Transformer 和 GAN 结合的多变量时间序列无监督异常检测模型. 它允许时间序列数据的重构, 展示 GAN 如何有效地应用于多变量时间序列异常检测, 还探索对鉴别器应用

patch 技巧的方法来捕捉时间窗口当中的异常细节特征. 本文还提出联合重构误差和鉴别误差来计算时间窗口的异常分数. 同时, 为了提高模型在异常事件发生时检测和解释异常的诊断能力, 引入图注意力层来对许多具有潜在相互关系的传感器数据进行建模. 在 3 个真实世界时间序列数据集上的实验表明, 所提方法在大多数情况下优于基线方法, 进一步分析表明, 该方法具有良好的可解释性, 可用于异常诊断. 未来目标是更多地探索将这种方法与额外的架构结合起来, 以进一步提高该方法的实用性.

参考文献

1 Oliveira J C M, Pontes K V, Sartori I, et al. Fault detection and diagnosis in dynamic systems using weightless neural networks. Expert Syst Appl, 2017, 84: 200–219

2 Li J, Izakian H, Pedrycz W, et al. Clustering-based anomaly detection in multivariate time series data. Appl Soft Computing, 2021, 100: 106919

3 Jin B, Chen Y, Li D, et al. A one-class support vector machine calibration method for time series change point detection. In: Proceedings of IEEE International Conference on Prognostics and Health Management (ICPHM), 2019. 1–5

4 Ishimtsev V, Bernstein A, Burnaev E, et al. Conformal k-NN anomaly detector for univariate data streams. In: Proceedings of Conformal and Probabilistic Prediction and Applications, 2017. 213–227

5 Xu D, Wang Y, Meng Y, et al. An improved data anomaly detection method based on isolation forest. In: Proceedings of the 10th International Symposium on Computational Intelligence and Design (ISCID), 2017. 287–291

6 Munir M, Siddiqui S A, Dengel A, et al. DeepAnT: a deep learning approach for unsupervised anomaly detection in time series. IEEE Access, 2018, 7: 1991–2005

7 Hundman K, Constantinou V, Laporte C, et al. Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018. 387–395

8 Su Y, Zhao Y, Niu C, et al. Robust anomaly detection for multivariate time series through stochastic recurrent neural network. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019. 2828–2837

9 Park P, Marco P D, Shin H, et al. Fault detection and diagnosis using combined autoencoder and long short-term memory network. Sensors, 2019, 19: 4612

10 Li D, Chen D, Jin B, et al. MAD-GAN: multivariate anomaly detection for time series data with generative adversarial networks. In: Proceedings of International Conference on Artificial Neural Networks, 2019. 703–716

11 Bashar M A, Nayak R. TAnoGAN: time series anomaly detection with generative adversarial networks. In: Proceedings of IEEE Symposium Series on Computational Intelligence (SSCI), 2020. 1778–1785

12 Geiger A, Liu D, Alnegheimish S, et al. TadGAN: time series anomaly detection using generative adversarial networks. In: Proceedings of 2020 IEEE International Conference on Big Data (Big Data), 2020. 33–43

13 Audibert J, Michiardi P, Guyard F, et al. USAD: unsupervised anomaly detection on multivariate time series. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2020. 3395–3404

14 Jiang W, Hong Y, Zhou B, et al. A GAN-based anomaly detection approach for imbalanced industrial time series. IEEE Access, 2019, 7: 143608

15 Zhou H, Zhang S, Peng J, et al. Informer: beyond efficient transformer for long sequence time-series forecasting. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2021. 11106–11115

16 Wu N, Green B, Ben X, et al. Deep transformer models for time series forecasting: the influenza prevalence case. 2020. ArXiv:200108317

17 Cook A A, Mısırlı G, Fan Z. Anomaly detection for IoT time-series data: a survey. IEEE Internet Things J, 2019, 7: 6481–6494

18 Zong B, Song Q, Min M R, et al. Deep autoencoding Gaussian mixture model for unsupervised anomaly detection. In: Proceedings of International Conference on Learning Representations, 2018

19 Gong D, Liu L, Le V, et al. Memorizing normality to detect anomaly: memory-augmented deep autoencoder for unsupervised anomaly detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019. 1705–1714

20 Zhou Y, Qin R, Xu H, et al. A data quality control method for seafloor observatories: the application of observed time series data in the east china sea. Sensors, 2018, 18: 2628

21 Karasu S, Altan A. Recognition model for solar radiation time series based on random forest with feature selection approach. In: Proceedings of the 11th International Conference on Electrical and Electronics Engineering (ELECO), 2019. 8–11

22 Zhao P, Chang X, Wang M. A novel multivariate time-series anomaly detection approach using an unsupervised deep neural network. IEEE Access, 2021, 9: 109025–109041

23 Cholakov R, Kolev T. Transformers predicting the future. Applying attention in next-frame and time series forecasting. 2021. ArXiv:2108.08224

24 Zerveas G, Jayaraman S, Patel D, et al. A transformer-based framework for multivariate time series representation learning. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021. 2114–2124

25 Chen Z, Chen D, Zhang X, et al. Learning graph structures with transformer for multivariate time-series anomaly detection in IoT. IEEE Internet Things J, 2021, 9: 9179–9189

26 Wu S, Xiao X, Ding Q, et al. Adversarial sparse transformer for time series forecasting. In: Proceedings of Advances in Neural Information Processing Systems, 2020. 17105–17115

27 Mogren O. C-RNN-GAN: continuous recurrent neural networks with adversarial training. 2016. ArXiv:1611.09904

28 Yoon J, Jarrett D, van der Schaar M. Time-series generative adversarial networks. In: Proceedings of Advances in Neural Information Processing Systems, 2019

29 Schlegl T, Seeböck P, Waldstein S M, et al. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: Proceedings of International Conference on Information Processing in Medical Imaging, 2017. 146–157

30 Akcay S, Atapour-Abarghouei A, Breckon T P. GANomaly: semi-supervised anomaly detection via adversarial training. In: Proceedings of Asian Conference on Computer Vision, 2018. 622–637

31 Jiang Y, Chang S, Wang Z. TransGAN: two pure transformers can make one strong GAN, and that can scale up. In: Proceedings of Advances in Neural Information Processing Systems, 2021. 14745–14758

32 Lin T, Wang Y, Liu X, et al. A survey of transformers. 2021. ArXiv:2106.04554

33 Zhou J, Cui G, Hu S, et al. Graph neural networks: a review of methods and applications. AI Open, 2020, 1: 57–81

34 Sanchez-Lengeling B, Reif E, Pearce A, et al. A gentle introduction to graph neural networks. Distill, 2021, 6: e33

35 Veličković P, Cucurull G, Casanova A, et al. Graph attention networks. 2017. ArXiv:1710.10903

36 Brody S, Alon U, Yahav E. How attentive are graph attention networks? 2021. ArXiv:2105.14491

37 Deng A, Hooi B. Graph neural network-based anomaly detection in multivariate time series. In: Proceedings of the AAAI Conference on Artificial Intelligence, 2021. 4027–4035

38 Zhao H, Wang Y, Duan J, et al. Multivariate time-series anomaly detection via graph attention network. In: Proceedings of IEEE International Conference on Data Mining (ICDM), 2020. 841–850

39 Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need. In: Proceedings of Advances in Neural Information Processing Systems, 2017

40 Beltagy I, Peters M E, Cohan A. Longformer: the long-document transformer. 2020. ArXiv:2004.05150

41 Liew S S, Khalil-Hani M, Bakhteri R. Bounded activation functions for enhanced training stability of deep neural networks on visual pattern recognition problems. Neurocomputing, 2016, 216: 718–734

42 Arjovsky M, Chintala S, Bottou L. Wasserstein generative adversarial networks. In: Proceedings of International Conference on Machine Learning, 2017. 214–223

43 Gulrajani I, Ahmed F, Arjovsky M, et al. Improved training of Wasserstein GANs. In: Proceedings of Advances in Neural Information Processing Systems, 2017

44 Xu H, Chen W, Zhao N, et al. Unsupervised anomaly detection via variational auto-encoder for seasonal KPIs in web applications. In: Proceedings of the 2018 World Wide Web Conference, 2018. 187–196

45 Isola P, Zhu J-Y, Zhou T, et al. Image-to-image translation with conditional adversarial networks. In: Proceedings of

the IEEE Conference on Computer Vision and Pattern Recognition, 2017. 1125–1134
46 Siffer A, Fouque P-A, Termier A, et al. Anomaly detection in streams with extreme value theory. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017. 1067–1075
47 Defreitas A, Alexander W N, Devenport W, et al. Anomaly detection in wind tunnel experiments by principal component analysis. AIAA J, 2022, 60: 2297–2307
48 Reddy D K K, Behera H S, Pratyusha G M S, et al. Ensemble bagging approach for iot sensor based anomaly detection. In: Proceedings of Intelligent Computing in Control and Communication. Singapore: Springer, 2021. 647–665
49 Park D, Hoshi Y, Kemp C C. A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder. IEEE Robot Autom Lett, 2018, 3: 1544–1551

Transformer-GAN architecture for anomaly detection in multivariate time series

Meiling CAI[1], Jiaxi WANG[1], Jinping LIU[1,2*], Zhaohui TANG[3] & Yongfang XIE[3]

1. *Hunan Provincial Key Laboratory of Intelligent Computing and Language Information Processing, Hunan Normal University, Changsha* 410081*, China*;
2. *Key Laboratory of Computing and Stochastic Mathematics* (*Ministry of Education*)*, Hunan Normal University, Changsha* 410081*, China*;
3. *School of Automation, Central South University, Changsha* 410083*, China*
* Corresponding author. E-mail: ljp@hunnu.edu.cn

Abstract Anomaly detection based on multivariate time series correlation data collected in real time during the process is one of the key aspects of preventing industrial process accidents and ensuring system safety. However, industrial multivariate time series anomaly detection still faces two major challenges: (1) the complex nonlinear correlation characteristics among time series data variables lack an effective representation method, and (2) the complex correlation among time series with highly unbalanced normal/abnormal distribution needs to be deeply explored. In this paper, we propose a Transformer generative adversarial networks (GAN)-based multivariate time series anomaly detection method (TGAN-MTSAD). TGAN-MTSAD employs transformer neural networks as the base model of GAN and introduces a graph attention layer to automatically learn complex dependencies among time series multivariate variables. It applies a patch trick to enable the model to effectively capture anomaly details within a time window. An anomaly score calculation method is proposed based on a combination of reconstruction and discrimination errors. An extensive performance validation and comparative experimental analysis of the proposed method were carried out using three real-world datasets. The results show that TGAN-MTSAD can effectively detect in-process timing anomalies, outperforming the baseline method in most cases, and has good interpretability for complex industrial anomaly detection.

Keywords multivariate time series, anomaly detection, Transformer, anomaly score, graph attention