计 算 机 研 究 与 发 展
Journal of Computer Research and Development
https://crad.ict.ac.cn
62(12): 2962−2973, 2025

基于多窗口划分集成学习的多维时间序列异常检测

王泽南[1] 王意洁[1] 周小晖[1] 熊旭东[2]
[1](并行与分布计算全国重点实验室(国防科技大学) 长沙 410073)
[2](国防科技大学计算机学院 长沙 410073)
(wangzenan19@nudt.edu.cn)

Anomaly Detection for Multivariate Time Series Based on Multi-Window Segmentation Ensemble Learning

Wang Zenan[1], Wang Yijie[1], Zhou Xiaohui[1], and Xiong Xudong[2]
[1](*National Key Laboratory of Parallel and Distributed Computing* (*National University of Defense Technology*), *Changsha* 410073)
[2](*College of Computer*, *National University of Defense Technology*, *Changsha* 410073)

Abstract In the era of large language models, the training and inference of large language models need the support of arithmetic resources, in which the anomaly detection of arithmetic resource data can effectively guarantee the training and inference of large language models. As the parameters of the large language model increase, the scale of the arithmetic resources used by the large language model grows, in which the data of multiple types of metrics reflecting the operating state of computility show more complex temporal changes over time. Existing multivariate time series anomaly detection methods typically use a preset window size to perform sliding slicing on multivariate time series data. However, a unified window that ignores the periodic characteristics of different dimensions may truncate the complete periodic patterns of time series data in some dimensions, hindering the anomaly detection model from learning the normal patterns of multivariate time series data and resulting in poor anomaly detection performance. To address this issue, we propose an unsupervised multivariate time series anomaly detection method SELAD based on ensemble learning with multi-window extraction. Specifically, we first extract the periodic patterns of each dimension in the multivariate time series data based on the Fourier frequency method, and then perform multi-window extraction to preserve the complete periodic patterns of each dimension. In the process of model training, the huge number of parameters of the large language model can solve the problem that the traditional model has a memory bottleneck when the sliding window increases, which leads to the deterioration of the learning effect. Subsequently, by designing a mixed expert models (MoEs), the time series data from multiple partitioned windows are input into an ensemble learning framework that integrates large language models and LSTM models for training, in order to learn and identify the normal temporal patterns of each dimension. Finally, anomaly detection is performed based on reconstruction scores. In this study, experiment results on four real-world multivariate time series datasets demonstrate that SELAD improves the *F*1 score by 17.87% to 90.77% compared with existing methods.

Key words multivariate time series; unsupervised anomaly detection; multi-window extraction; ensemble learning;

收稿日期: 2024-06-24; **修回日期**: 2025-04-09

基金项目: 新一代人工智能国家科技重大专项项目(2022ZD0115302); 国家自然科学基金项目(61379052); 国家教育部科研创新基金项目(2018A02002); 湖南省自然科学杰出青年基金项目(14JJ1026)

This work was supported by the National Science and Technology Major Project (2022ZD0115302), the National Natural Science Foundation of China (61379052), the Science Foundation of Ministry of Education of China (2018A02002), and the Natural Science Foundation for Distinguished Young Scholars of Hunan Province (14JJ1026).

通信作者: 王意洁(wangyijie@nudt.edu.cn)

large language model

摘　要　大语言模型时代下，大语言模型的训练和推理需要算力资源的支撑，其中针对算力资源指标数据的异常检测能够有效保障大语言模型的正常训练和推理. 随着大语言模型参数的增加，大语言模型使用的算力资源规模日益扩大，其中反映算力运行状态的多类指标数据随着时间推移呈现出更复杂的时序周期性变化. 现有的多维时序异常检测方法通常采用预设的窗口大小对多维时序数据进行滑动切片，但忽略不同维度周期特性的统一窗口划分会截断部分维度时序数据的完整周期性模式，阻碍了异常检测模型对多维时序数据正常模式的学习，导致异常检测效果变差. 为了解决这一问题，提出了一种基于多窗口划分集成学习的无监督多维时序异常检测方法 SELAD. 具体地，首先利用傅里叶变换提取多维时序数据中各维度的周期性模式，并鉴于此信息进行多窗口划分，以保留各维度数据的完整周期性模式. 在模型训练过程中，大语言模型参数量巨大的特点可以解决传统模型在滑动窗口增大后出现记忆瓶颈导致的学习效果变差的问题. 通过设计一种混合专家模型将保留完整周期划分的时序数据输入融合大模型和 LSTM 模型的集成学习框架进行训练，以学习并重构各维度的正常时序模式. 最终，基于重构误差检测多维时间序列数据中的异常. 通过在 4 个多维时间序列数据集上实验，SELAD 在平均 $F1$ 分数上相比现有方法提升了 17.87~90.77 个百分点.

关键词　多维时间序列；无监督异常检测；多窗口划分；集成学习；大语言模型

中图法分类号　TP391

DOI: 10.7544/issn1000-1239.202440573　　**CSTR:** 32373.14.issn1000-1239.202440573

随着大语言模型技术的发展（以下简称大模型），模型参数规模从数千亿扩展到数万亿，导致计算需求急剧增加，远超单颗 AI（artificial intelligence）芯片性能的增长速度. 根据大模型的尺度定律（scaling law），模型性能的提高与模型的参数和训练数据的大小都呈现幂律关系，进而需要更强的计算能力以支撑大模型训练. 因此，无论是通信运营商、互联网企业、AI 研发公司，都通过自建或使用规模化的计算中心进行大模型训练和推理. 当前，智算中心正处于建设阶段，中心内 GPU 显卡数量的非线性增加给大模型的训练带来了更高的不稳定性和协作难度[1]. OpenAI 公司使用 1 024 个 NVIDIA A100 GPU 训练 GPT-3 时，大约需要 34 天完成. 随着模型规模的扩大，以及训练节点数量的增加，训练时间也会延长，且节点在训练中发生故障的可能性也会更高. 据 GPU 训练集群的观察显示[2]，单张 GPU 在 1 个月内的故障率约为 8%，而每天的故障率大约为 0.27%. 对于一个有上千张 GPU 参与的训练作业来说，由于 GPU 故障，1 天内造成训练失败的概率可能高达 93%. 在此背景下，如何确保大模型训练的长期稳定性已成为一个关键的研究领域. 智能计算运维技术正在探索和发展的阶段，并被视为构建具备高容错和高效能的智能计算平台的关键技术. 这一技术领域涉及故障分析、数据预测、异常检测等多个方面[3]. 其中，多维时间序列异常检测技术是智能计算运维的重要研究方向之一，它能够识别和预警智能计算环境中复杂时序数据的异常点. 异常检测技术用于识别出与数据主体行为显著不符的记录，这些偏差可能由多种内外因素引起，且表现形式多样. 传统方法依赖领域专家手动设定静态阈值来界定异常，并据此报告潜在故障，然而，随着智能计算技术的快速发展，传统人工阈值设置及异常报告机制难以应对运维数据量级急剧膨胀，数据结构复杂度日益增大，不仅效率低下，还高度依赖人工介入，成本高昂. 近年来，受益于大数据分析和深度学习技术的快速发展，智能异常检测方案应运而生，逐步取代传统人工作业，在处理大量时间序列数据方面展现出显著优势. 然而，随着智算中心规模的不断扩大，算力资源的指标数据随着时间推移呈现出较强的关联性. 因此，在复杂的智能计算系统中，多维时间序列异常检测方法被广泛应用于更精确地检测算力资源的异常情况.

随着算力资源规模越来越大，智算中心产生的数据量巨大，而异常情况则是少数. 由于异常样本稀少，难以收集全面的异常标签数据用于监督学习. 同时，无监督学习通过学习正常行为的模式来识别偏离常态的行为，当系统遇到新的或未知类型的异常时，无监督模型依然能够有一定的识别能力，具有更好的自适应性和灵活性. 因此，无监督多维时间序列异常检测比有监督异常检测方法以及半监督异常检测方法更适用于智算系统的智能运维. 然而，在无监

督多维时间序列异常检测领域，现有的多维时序异常检测方法通常采用预设的窗口大小对多维时序数据进行滑动切片，但忽略不同维度周期特性的统一窗口划分会截断部分维度时序数据的完整周期性模式，阻碍了异常检测模型对多维时序数据正常模式的学习，导致异常检测效果差.

多维时间序列数据通常包含多个相互关联且具有不同特征的维度，现有的多维时序异常检测方法在进行窗口划分时没有考虑到各维度周期性模式的差异. 针对这一问题，本文提出了基于多窗口划分集成学习异常检测方法 SELAD(unsupervised anomaly detection in multivariate time series based on multi-window segmentation ensemble learning)，该方法设计一种多窗口划分机制对多维时序数据进行傅里叶变换分析得到每一维度相关的周期性模式，进行保留完整周期性模式的窗口划分. 然后，通过一种混合专家模型，将保留完整周期性模式划分的多维时序数据输入融合大模型和 LSTM 模型的集成学习框架进行训练，通过大模型和 LSTM 模型重构不同划分时序数据的正常时序模式，基于重构误差进行多维时间序列异常检测.

本文的主要贡献包括 3 个方面：

1)不同于现有方法对时间序列数据进行统一窗口划分，多窗口划分机制采用傅里叶分析方法捕获多维时序数据各自的周期特性，由此构建针对不同维度数据时序变化的自适应窗口.

2)提出一种基于集成学习的异常检测框架，采用包括大模型和 LSTM 模型的混合专家模型，通过利用大模型对长划分窗口时序数据的重建能力和 LSTM 模型对时间序列数据的学习能力来提高异常检测性能.

3)在 4 个多维时间序列数据集上进行了对比实验. SELAD 方法的平均 $F1$ 分数为 0.915 3，相较于现有方法提升了 17.87%~90.77%. 进一步地，通过消融实验证明了 SELAD 方法中多窗口划分机制能够保留多维数据的完整周期性模式，同时，包含大模型和 LSTM 模型的混合专家模型可以更好地学习不同划分窗口时序数据的正常时序模式.

1 相关工作

目前，无监督多维时序异常检测方法主要分为 2 类：基于深度学习的异常检测方法和基于大模型的异常检测方法[4].

1.1 基于深度学习的异常检测方法

基于深度学习的异常检测方法使用深度学习模型学习多维时间序列数据的正常模式来进行异常检测. 主要包括基于重构的异常检测方法和基于预测的异常检测方法.

基于重构的异常检测方法通过学习多维时间序列数据的编码模式来重构出多维时间序列数据，通过比较重构数据和原始数据的差别来进行异常检测. Hochreiter 等人[5]提出了一种利用长短期记忆网络(LSTM)的编码器-解码器方法，称为 EncDec-AD. 该方法通过学习和重建正常时间序列的行为模式，并利用重建时产生的误差来有效识别和检测时间序列中的异常现象. Park 等人[6]提出了一种多模态异常检测方法 LSTM-VAE，该方法通过结合 LSTM 和变分自编码器(VAE)[6]，能够有效地整合多种信号并重建它们的预期分布. Audibert 等人[7]提出了一种名为 USAD 的无监督异常检测方法，该方法是基于对抗性训练的自编码器. USAD 的自编码器架构允许其以无监督的方式进行学习. 通过对抗性训练和特定架构，USAD 能够在放大异常输入的重建误差的同时，实现快速训练. Zhang 等人[8]提出了一种创新的无监督异常检测方法，名为具有自监督学习的自适应记忆网络(AMSL)，该方法基于卷积自编码器结构，并通过自监督学习模块来学习通用的正常模式，以及通过一个自适应记忆融合模块来学习丰富的特征表示. Campos 等人[9]提出了一种新颖的无监督时间序列异常检测方法，称为基于多样性驱动的卷积集成(CAE-ensemble). 该方法利用基于卷积序列到序列自编码器构建多个基础异常检测模型，这些模型能够捕捉时间序列中的时序依赖性. 此外，还引入了一种新颖的多样性驱动训练方法，以保持基础模型间的多样性，目的是提高集成模型的准确性. 为了提高效率，该方法在训练过程中支持高度的并行性，并且能够通过一种基于“再生网络”的迁移学习方法，在不同的基础模型之间转移一些模型参数，从而减少训练时间. Su 等人[10]提出了 OmniAnomaly 方法，用于检测多维时间序列中的异常. 该模型通过学习时间序列数据的上下文信息，并使用平面归一化流识别其背后的正常模式，将具有较低重建概率的模式识别为异常[3].

基于预测的异常检测方法通过学习多维时间序列数据的时序趋势，根据点或最近的窗口来预测点或子序列，通过比较预测值和实际值来确定异常. Hundman 等人[11]提出了一种基于 LSTM 的异常检测方法 LSTM-NDT. 该方法利用 LSTM 网络的对时间序列

的学习能力，对时间序列数据进行预测. 通过学习正常操作条件下的行为模式，LSTM 能够识别出与训练数据显著不同的异常模式. Wu 等人[12]针对大部分时间序列数据具有复杂的季节性特征的问题，提出了一种基于预测的无监督异常检测方案 AD-LTI，采用了一个结合时间序列数据分解和推断的骨干模型. 此外，该方法提出了一种新颖的度量，即局部趋势不一致性（LTI），以及一种高效的检测算法，可以实时计算 LTI，并根据其异常概率稳健地对每个数据点进行评分. Deng 等人[13]认识到现有方法没有明确地学习变量之间现有关系的结构，或者利用它们来预测时间序列的预期行为. 提出了一种将结构学习方法与图神经网络相结合的方法 GDN，使用注意力权重来解释检测到的异常.

然而，不论是基于重构的深度学习方法还是基于预测的深度学习方法，都没有考虑到统一窗口划分会截断部分维度时序数据的完整周期性模式，忽略了不同维度的周期特性，从而阻碍了深度学习异常检测模型对多维时序数据正常模式的学习.

1.2 基于大模型的异常检测方法

目前使用大模型进行时序异常检测任务的主要方法分为 2 类：基于时序数据大模型的异常检测方法和基于自然语言大模型的时序异常检测[14].

时序数据大模型采用定制化模型训练方法，提高了模型对时序数据的理解能力[15]. 通过专业化训练，模型可以更好地理解时间序列的时序性质和正常模式. Dang 等人[16]基于 Transformer 的模型架构提出了 TimeGPT-1 模型，模型通过采集一定时间窗口内的历史数据作为输入，添加位置编码来帮助捕捉时序信息，通过编码器-解码器后，利用一个线性层将解码器输出映射至预测窗口维度，并最终在大规模数据集上训练来最小化预测误差. Chen 等人[17]提出了开源的大型预训练时间序列模型 MOMENT. 通过收集大量的公共时间序列数据，并将其整合成“时间序列堆栈”（time-series pile），然后利用这些数据在遮蔽时间序列预测任务上预训练一个 Transformer 模型. Zhou 等人[18]提出了一个简单有效的预训练概率时间序列模型框架 Chronos. 通过使用缩放和量化技术将时间序列值标记为固定词汇，并通过交叉熵损失在这些标记化的时间序列上训练现有的基于 Transformer 的语言模型架构.

大模型使用多种文本数据（包括时间序列数据[19]）进行预训练，并且使用少量的示例就可以在时间序列异常检测任务中表现出色. Khlaisamniang 等人[20]提出了一种基于自然语言处理模型 BERT 来检测监测数据中的异常. 该方法只需要少量的标签数据来训练 BERT 模型，就能获得比现有方法更好的结果. Garza 等人[21]提出了一种将日志序列视为自然语言序列的方法 BERT-Log. 该方法使用预训练的语言模型来学习正常和异常日志的语义表示. 然后使用一个完全连接的神经网络来微调 BERT 模型以检测异常. Sarda 等人[19]提出了一种冻结预训练 Transformer 的方法 OFA（one fits all）. 通过从数十亿数据中预训练语言模型来解决大模型在时间序列分析中缺少大量用于训练的数据的问题. 同时，避免改变预训练模型中的自注意力和前馈层来对时间序列数据进行重构，通过评估重构误差来对时间序列进行异常检测. Goswami 等人[22]提出了一种名为 ADARMA 的基于大模型的平台来进行微服务平台部署的实时异常检测，通过异常检测减少了停机时间，进而提高微服务平台的系统可靠性. Ansari 等人[23]通过将生成式人工智能技术集成到异常检测修复系统中，通过使用 GPT-4 进行自动代码生成来进行异常的修复，从而提高大规模系统运行的稳定性.

在基于大模型的异常检测方法中，虽然大模型的参数量巨大，可以很好地学习时间序列数据中的长时间依赖，但是大模型的训练时间过长，以及需要的数据集庞大是阻碍大模型在时间序列异常检测领域的关键挑战.

2 基于多窗口划分集成学习的无监督多维时序异常检测方法

2.1 问题定义

首先，多维时间序列数据为等间隔采样的连续观测值序列. 定义长度为 T 的输入时间序列 $\boldsymbol{X}=(\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_T)$，其中 $\boldsymbol{X}\in\mathbb{R}^{D\times T}$，$D$ 表示观测值的维度数，T 表示多维时间序列的长度. 异常可以发生在任何时间点 $t\in\{1,2,\cdots,T\}$ 的任何维度 $d\in\{1,2,\cdots,D\}$ 中. 在多维时间序列异常检测任务中，对于输入的时间序列 $\boldsymbol{X}$，需要输出对应的异常标签序列 $\boldsymbol{Y}=(y_1,y_2,\cdots,y_T)$，其中 $y_t\in\{0,1\}$，对于任何时间点 $t\in\{1,2,\cdots,T\}$. $y_t=0$ 表示时间点 t 为正常，$y_t=1$ 表示时间点 t 为异常.

2.2 数据预处理

在多维时间序列数据中，由于时间序列具有连续性，不同时间点的数据相互依赖. 因此，在进行异常检测时，需要考虑时间观测点 $\boldsymbol{x}_t$ 与其历史时间点之间的依赖关系. 因此，使用滑动窗口方法来处理多

维时间序列异常检测任务, 构建了一个长度为 N 的时间窗口 $\boldsymbol{w}_t$, 定义为

$$\boldsymbol{w}_t = (\boldsymbol{x}_{t-N+1}, \boldsymbol{x}_{t-N+2}, \cdots, \boldsymbol{x}_t). \tag{1}$$

对于时间序列中的时间观测点 $\boldsymbol{x}_t$, 如果 $t<N$, 由于缺少足够的历史数据来构建长度为 N 的时间窗口, 无法评估其异常性. 为了避免由于时间窗口太短而对模型的正常模式重构造成影响, 将这些时间点 $\boldsymbol{x}_t$ 视为正常[3]. 为了更好地提升模型学习数据正常模式的能力, 解决统一窗口会截断部分维度时序数据完整周期性模式的问题, 在多维时间序列异常检测模型训练前需要对多维时间序列数据进行时间周期性分析.

定义 1. 傅里叶变换. 时间序列 T 的长度为 n, 对时间序列 T 进行离散傅里叶变换, 得到一系列复杂系数 $\boldsymbol{C}=(c_0, c_1, \cdots, c_{n-1}) \in \mathbb{C}^n$, 相应的一系列频率被定义为 $\boldsymbol{F} = \left(f_0, f_1, \cdots, f_{\left\lceil \frac{n-1}{2} \right\rceil}\right)$.

对于多维时序数据 $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_T)$, 其中每个 $\boldsymbol{x}_t$ 是一个 D 维向量. 首先对每一维数据进行傅里叶变换:

$$\boldsymbol{X}_d = (x_{d,1}, x_{d,2}, \cdots, x_{d,T}), \tag{2}$$

其中 $d \in \{1,2,\cdots,D\}$, 对每一维的数据进行傅里叶变换, 得到傅里叶系数

$$\boldsymbol{C}_d = FFT(\boldsymbol{X}_d), \tag{3}$$

其中 $\boldsymbol{C}_d$ 中每个元素 $C_{d,k}$ 对应频率 F_k 的幅度和相位. 对于每个维度 d, 我们感兴趣的是每个频率 F_k 的幅度, 可以通过计算傅里叶系数的模量来获得:

$$\left|C_{d,k}\right| = \sqrt{Re(C_{d,k})^2 + Im(C_{d,k})^2}. \tag{4}$$

对于每个维度 d, 我们选择具有最大幅度 $\left|C_{d,k}\right|$ 的频率 F_k 作为该维度的主要频率:

$$F_{d,\max} = \arg\max_k \left|C_{d,k}\right|, \tag{5}$$

其中每个维度 d 的划分窗口大小为

$$m_d = \frac{1}{F_{d,\max}}. \tag{6}$$

然而, 每一维时间序列数据的周期性模式并不相同, 为了避免窗口切分后数据长度的不一致性, 需要在划分前进行数据补全.

对于不同的划分窗口 $M=\{m_1, m_2, \cdots, m_D\}$, 划分后的数据长度最长为 $T-\min(M)+1$. 并在每一维数据前补齐缺少的数据(对于第 k 维, 需要补全的数据长度为: $m_k-\min(M)$, 通常采用补 0 的方式进行数据补全), 从而保证数据维度窗口划分长度的一致性. 由于会将小于窗口大小的前 N 个时间观测点视为正常时间点, 对每一维进行数据补齐操作并不影响多维时间序列异常检测效果.

2.3 集成学习模型

SELAD 方法是一个基于多窗口划分集成学习的无监督多维时序异常检测方法. 如图 1 所示, 模型整体是由多维时序窗口划分模块、混合专家(MoE)模块和包含大模型和 LSTM 模型的集成学习模块 3 部分组成. 首先, 对于输入的时间序列数据 $\boldsymbol{X}$ 的每一维数据进行傅里叶变换, 得到每一维中最主要频率的幅度 M; 随后, 根据不同的幅度大小对每一维进行滑动窗口划分, 得到不同划分的时间序列数据; 接着通过混合专家模型中定义的门控网络来对划分后的时间序列数据进行模型的选择; 最后通过集成学习框架对时间序列数据进行重构, 和输入的时间序列数据 $\boldsymbol{X}$ 进行重构误差的计算来得到异常标签序列 $\boldsymbol{Y}$.

通常来说, 在多维时间序列数据中, 不同维度数据之间呈现出相关性特征. 然而, 在实际应用中, 少

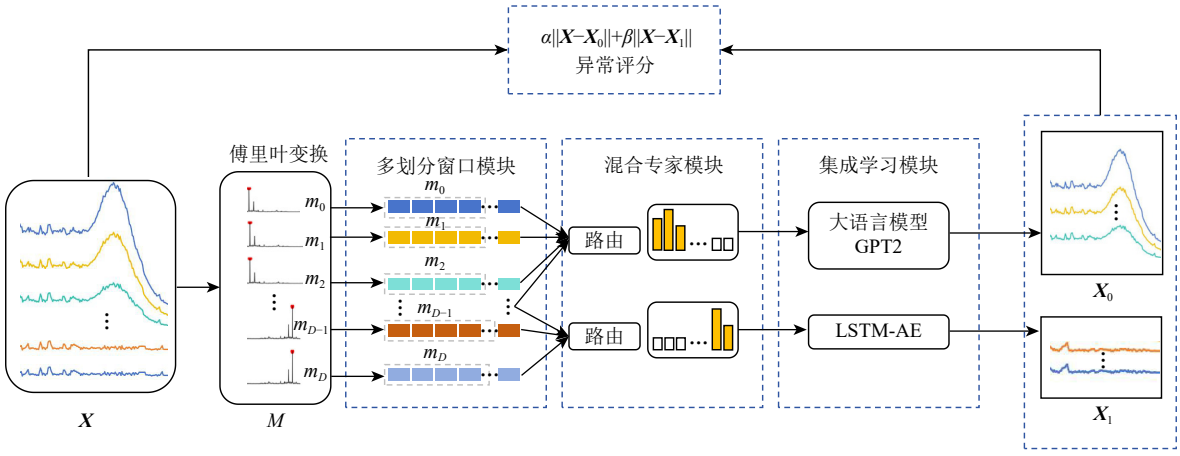


Fig. 1 The overall architecture of SELAD

图 1 SELAD 整体架构

数维度数据特征和主体特征之间存在差异，对于模型训练来说，统一的窗口划分会导致模型无法精准地捕获数据中的特征. 首先，将多维时间序列数据 T 作为框架的输入，通过多维数据中每一维进行傅里叶变换，精准得到每一维的时间序列数据的主要频率. M 是包含了每一维时间序列数据主要频率的幅度的序列，使用 $M=\{m_1, m_2, \cdots, m_D\}$ 来表示. 通过进行数据补全，将多维时间数据划分成长度一致的不同划分窗口的子数据.

由于傅里叶变换是一个非参数方法，对于不同的多维时间序列数据，通过傅里叶变换得到的各维度主要傅里叶频率序列 M 并不相同，固定的分配策略并不能进行划分窗口不同的情况下集成模型的训练. 因此，设计了一种针对不同划分的多维时间序列数据的混合专家模型来解决滑动窗口非参数化带来的模型训练问题.

首先，定义了索引矩阵 $\boldsymbol{I}$ 来表示从时间序列数据到专家的选择标记，其中 $I[i, j]$ 表示第 i 个专家选择第 j 个时间序列数据的标记. 其次，定义了门控矩阵 $\boldsymbol{G}$ 表示所选时间序列数据的专家权重. 这些矩阵是通过门控网络进行学习. 最后通过矩阵 $\boldsymbol{P}$ 对输入的时间序列数据进行处理，矩阵 $\boldsymbol{P}$ 是矩阵 $\boldsymbol{I}$ 的 one-hot 编码版本，得到各个专家需要的输入数据.

$$\boldsymbol{S} = Softmax(\boldsymbol{T} \times \boldsymbol{W}_g), \tag{7}$$

其中 $\boldsymbol{S}$ 表示划分后的时间序列数据与专家之间的亲和分数.

$$\boldsymbol{G} = \boldsymbol{I} = TopK(\boldsymbol{S}^{\mathrm{T}}, k), \tag{8}$$

$$\boldsymbol{P} = Onehot(\boldsymbol{I}). \tag{9}$$

对于已经划分好的时间序列输入 $\boldsymbol{X}$ 来说，通过对矩阵 $\boldsymbol{P}$ 和时间序列 $\boldsymbol{X}$ 的向量积得到每个专家的输入集合. 随后通过大模型和 LSTM 模型的混合专家模型 MoE，得到重构后的时间序列 $\boldsymbol{X}_{\mathrm{out}}$.

$$\boldsymbol{X}_{\mathrm{in}} = \boldsymbol{P}, \tag{10}$$

$$\boldsymbol{X}_{\mathrm{out}} = MoE(\boldsymbol{X}_{\mathrm{in}}, \boldsymbol{G}). \tag{11}$$

对于集成学习模型，SELAD 方法使用了大模型 GPT-2 和 LSTM 模型进行集成学习模型的构建. 本方法通过使用在自然语言处理领域中预训练好的 GPT-2 模型进行模型迁移训练，通过迁移训练可以减少任务特定的重新训练需求. 为了将自然语言处理领域的预训练模型应用于时间序列这种新模态中，需要将时间序列数据投影到预训练所需要的维度上去. 从而，大模型可以学习到多维时间序列数据中的正常模式并进行重构. 同时，SELAD 方法使用多窗口划分机制，划分后的子数据存在划分窗口不一致的问题. 为了避免子数据的形状影响模型的训练，采用 LSTM 模型可以不考虑输入数据的划分窗口大小，动态适应不同长度的输入数据，将不同划分的子序列通过输入到 LSTM 模型中学习输入多维时间序列数据的正常模式.

通过重构误差损失设计模型的损失函数，旨在最小化原始输入时间序列与模型重构时间序列之间的差异. 在模型训练阶段，集成学习模型对多维时间序列输入 $\boldsymbol{X}$ 进行重构，重构结果用 $\boldsymbol{X}_0$ 和 $\boldsymbol{X}_1$ 表示，定义模型的损失函数为：

$$L = \alpha \|\boldsymbol{X} - \boldsymbol{X}_0\| + \beta \|\boldsymbol{X} - \boldsymbol{X}_1\|, \tag{12}$$

其中 α 和 β 是混合专家模型通过对主傅里叶频率幅度序列 M 进行分析得到的参数，$\alpha + \beta = 1$. 在模型推理阶段，将多维时间序列数据 $\boldsymbol{X}$ 输入到训练好的模型进行重构，得到 $\boldsymbol{X}_0$ 和 $\boldsymbol{X}_1$，然后通过模型的损失函数公式进行异常评分，异常评分越高表明该时间点越异常.

2.4 计算复杂度

大模型时代下，随着硬件能力的增强和深度学习需求的增长，模型规模（参数数量）显著扩大. 比如，从 AlexNet[24] 到 BERT[25] 等大模型，参数量呈指数增长，算法的计算复杂度和资源消耗越来越不可忽视.

SELAD 方法是一个基于多窗口划分集成学习的无监督多维时序异常检测方法. 在 SELAD 方法的多窗口划分模块中，多维时间序列数据作为框架的输入，通过对多维数据的每一维数据进行傅里叶变换，得到每一维的时间序列数据的主要频率. 定义多维时间序列数据的长度为 n，多维时间序列的维数为 d. 相比传统傅里叶变换的时间复杂度 $O(n^2)$，本方法采用快速傅里叶变换来提高计算效率，具体计算过程如下：快速傅里叶变换通过将一个长度为 n 的傅里叶变换分解为 2 个长度为 $n/2$ 的傅里叶变换进行计算，然后合并 2 个 $n/2$ 的傅里叶变换结果，需要 $O(n)$ 的操作. 每次分解将问题规模减半，递归深度为 $\log n$. 单维傅里叶变换的时间复杂度为 $O(n\log n)$，因此 SELAD 方法的多窗口划分模块的时间复杂度为 $O(dn\log n)$，空间复杂度为 $O(dn)$.

在 SELAD 方法的混合专家模块中，定义专家模型的数量为 E，每个专家模型的时间复杂度为 $O(T_{\mathrm{m}})$，门控网络的时间复杂度为 $O(T_{\mathrm{g}})$. 对于每个输入，混合专家模块只激活一小部分专家学习数据的上下文信息来降低时间复杂度开销，假设激活的专家数为 k.

因此混合专家系统的时间复杂度为 $O(kT_e+T_g)$. 虽然混合专家模块具有多个专家, 但每次训练无需同时加载所有专家, 实际存储开销不大. 混合专家模块通过动态选择专家来降低计算复杂度, 使得在模型能力与效能之间实现平衡.

在 SELAD 方法的集成学习模块中, SELAD 方法使用了大模型 GPT-2 和 LSTM-AE 模型进行集成学习模型的构建. GPT-2 模型是基于 Transformer 架构的大模型, 其时间复杂度和空间复杂度主要由自注意力机制和前馈神经网络部分决定. 计算注意力得分矩阵时, 需要计算所有序列位置的内积, 时间复杂度为 $O(n^2d)$. 在前馈神经网络中, 由于每个序列位置的特征维度可以独立计算, 但需要跨 2 个全连接层进行计算, 因此前馈神经网络的时间复杂度为 $O(nd^2)$. 对于 GPT-2 模型, 假设包含 L 层 Transformer, 因此 GPT-2 模型的总时间复杂度为 $O(L(n^2d+nd^2))$. Transformer 主要受输入序列的长度 n 和特征维度 d 影响, SELAD 采用的是周期性分割策略, 从更小的局部窗口中建模完整的上下文信息, 在规避信息截断的同时可以避免固定窗口长度带来的高复杂度影响. GPT-2 模型的空间复杂度受其参数量存储和中间激活值存储的影响. 参数量存储主要包括注意力机制的权重矩阵和前馈网络的权重矩阵, 其空间复杂度为 $O(Ld^2)$. 在前向传播中, 每层需要存储中间激活值, 以便在反向传播时使用, 空间复杂度为 $O(Lnd)$. 将参数和激活存储结合, 总的空间复杂度为 $O(L(d^2+nd))$. LSTM-AE 模型是一种基于 LSTM 的自编码器模型. 定义隐藏状态维度为 h, 每个时间步的计算涉及矩阵乘法和非线性激活函数, 时间复杂度为 $O(dh+h^2)$. 因此 LSTM-AE 模型的时间复杂度为 $O(n(dh+h^2))$.

3 实 验

3.1 实验设置

本实验将在 4 个时间序列数据集上对 SELAD 方法和 7 种流行异常检测方法进行测试, 对比分析 SELAD 方法的异常检测性能. 本实验环境采用 PyTorch 框架构建模型, 全部实验均在 Ubuntu 18.04 环境下进行. 在训练阶段, 设置批次训练的样本数为 128, 初始学习率设置为 0.000 5, 采用 ADAM 优化器对模型进行优化. 由多窗口划分模块通过傅里叶变换方法确定得到每一维的滑动窗口大小. 在实验中, 数据集被按照原始划分分为训练集和测试集, 并从训练集中抽取 25% 作为验证集. 为了确保实验结果的稳定性, 每个数据集上进行了 5 次重复实验, 并将这 5 次实验结果的平均值作为最终实验结果.

3.2 实验数据集

在实验数据集选择上, 本实验采用的是 4 个真实世界中的多维时间序列数据集. 主要选择了 1 个智能运维数据集 ASD, 该数据集包含了算力资源状态指标数据. 同时, 包括了 1 个大规模的多维服务器群组数据集 SMD 和 2 个传感器数据集 MSL 和 SMAP.

1)ASD 数据集[23]. 数据来源于一家大型互联网公司, 涉及 12 个描述服务器状态的实体. 数据集包含 45 天的多维时间序列数据和 19 个指标, 这些指标用于描述服务器状态, 包括 CPU、内存和网络相关的指标. 数据观测结果每 5 分钟记录 1 次. 前 30 天的数据作为训练集, 后 15 天则用于测试集[3].

2)SMD 数据集[6]. 该数据集收集了 28 台机器连续 5 周的数据, 每组数据之间的采集间隔为 1min. 每台机器收集了 38 个维度(变量)的信息. 数据集被分为训练集和测试集, 比例为 1∶1. 训练集是无标签的数据, 而测试集带有标签. 数据集中主要存在的异常包括点异常和上下文异常.

3)MSL 数据集[10]. 该数据集由 NASA 火星探测器的传感器和执行器的数据组合而成, 包含 55 个维度的传感器数据, 时间粒度为每分钟 1 次. 数据集中标记了不同类型的异常, 分为点异常和上下文异常, 分别对应突发性和趋势性变化.

4)SMAP 数据集[10]. 该数据集是由 NASA 火星探测器收集的航天器遥测流数据组成. 数据集包括 25 个维度的传感器信息, 其中包括点异常和上下文异常.

实验所用到的这 4 个数据集信息如表 1 所示.

Table 1 Dataset Statistics

表 1 数据集统计信息

数据集	训练数据条数	测试数据条数	维度
ASD	102 331	51 840	19
SMD	708 433	708 448	38
MSL	58 317	73 729	55
SMAP	135 183	427 617	25

3.3 评价指标

本文通过 3 个指标来评估模型性能, 分别是: $F1$ 分数、P-R 曲线下的面积(area under the precision-recall curve, AUC-PR)以及曲线下 ROC 面积(area under the receiver operating characteristic curve, AUC-ROC). 其中 P 为精确率(precision)、R 为召回率(recall).

$$P=\frac{TP}{TP+FP}, \tag{13}$$

$$R=\frac{TP}{TP+FN}, \tag{14}$$

$$F1=2\times\frac{P\times R}{P+R}. \tag{15}$$

TP(真正例)表示被正确地检测为异常的实际异常样本，*FP*(假正例)表示被错误地检测为异常的实际正常样本，*TN*(真负例)表示被正确识别为正常的实际正常样本，而 *FN*(假负例)表示被错误识别为正常的实际异常样本. AUC-PR 和 AUC-ROC 分别通过计算 *P*-*R*(precision-recall)曲线下的面积和 ROC(receiver operating characteristic)下的面积来获得[3].

3.4 对比方法

在本实验中，我们将 SELAD 与 2 种经典的多维时间序列异常检测方法、4 种深度学习多维时间序列异常检测方法及 1 种大模型进行比较，以验证其有效性.

1)iForest[25]. 孤立森林(isolation forest, iForest)是一种基于集合的异常检测方法. iForest 随机选择一个超平面来切割数据空间，每次切割都会将数据分成 2 个子空间，这一过程反复进行，直到每个子空间只剩下 1 个数据点. 稀疏分布并远离密集区域的数据点相比于高密度区域，需要更少的子空间切割. 通过多次随机切割，若某个数据点的平均切割次数低于某一阈值，则该点会被视为异常. iForest 算法由于其线性时间复杂度，非常适合用于大规模数据的异常检测.

2)COPOD[26]. COPOD 是一种基于 copula 进行异常检测的方法. copula 是一种概率统计函数，能够对多维累积分布进行建模，并有效地描述多个随机变量之间的依赖关系. COPOD 采用非参数方法，通过经验累积分布来估计各维度联合分布的尾部概率. 最终，通过评估偏离均值的尾部概率来识别异常. 该方法无需参数调整，计算开销低，并且在异常归因方面提供了一定的可解释性.

3)DSVDD[27]. DSVDD 是一种结合了自编码器(autoencoder)和支持向量数据描述器(SVDD)的深度学习异常检测方法. 该方法在训练阶段同时对自编码器的参数和其潜在表示中的包含超球体的体积进行最小化. DSVDD 提出了一种新颖的异常分数，该分数结合了自编码器的重构误差和样本与超球体中心在潜在表示中的欧氏距离. 通过最小化这个异常分数，DSVDD 能够在训练期间学习正常类别的底层分布.

4)LSTM-VAE[6]. LSTM-VAE 混合模型是一种结合了变分自编码器和 LSTM 的深度学习架构，用于时间序列的异常检测. 该模型首先利用 VAE 来学习时间序列数据的低维表示，并捕捉数据的潜在分布，从而获得重构误差. 随后，LSTM 网络利用这些潜在表示来建模时间序列的动态特性. 该模型结合了 VAE 的生成能力和 LSTM 的时序预测能力. 然而，VAE 部分可能会在捕捉复杂的时间序列分布时遇到困难，导致重构误差的准确性受限. 此外，LSTM 模型在处理非常长的时间序列或需要长期记忆的任务时可能会遇到梯度消失或爆炸的问题.

5)USAD[7]. USAD 是一种基于自编码器架构的无监督异常检测方法，通过对抗性训练启发式学习，使其能够在不需要标记数据的情况下进行训练. USAD 利用了生成对抗网络(GANs)的原理，通过 2 个解码器网络 D_1 和 D_2 共享同一个编码器网络 E 来构建其架构. 在第 1 阶段，USAD 训练 2 个自编码器 AE_1 和 AE_2 来学习重构正常输入窗口 W. 在第 2 阶段，USAD 采用对抗性训练策略，AE_1 试图欺骗 AE_2，而 AE_2 的目标是学习区分数据是真实的还是由 AE_1 重构的. 然而，其对抗性训练可能需要仔细的调整和监控，以避免常见的问题，如模式崩溃和非收敛性.

6)TS-CP2[28]. TS-CP2 是一种利用对比学习策略进行时间序列变点检测的方法，通过学习嵌入式表示来区分时间相邻间隔对的嵌入与跨时间分隔的间隔嵌入对. 这种方法首次将对比学习应用于变点检测问题，通过最大化连续时间窗口间的互信息来学习表示，同时利用对比损失函数来训练编码器，使得相邻时间窗口的嵌入对在嵌入空间中更接近，而非相邻时间窗口的嵌入对更远离.

7)OFA[18]. OFA(one fits all)是一种基于预训练模型 GPT-2 的无监督异常检测方法，其模型定义为 GPT4TS，用于时间序列的异常检测. 该方法的核心在于避免改变预训练语言模型或图像模型中残差块的自注意力和前馈层来保留来自预训练模型的大部分学习知识，以此来进行时间序列正常模式的学习.

3.5 对比结果分析

本研究通过在 4 个多维时间序列数据集上对 7 种多维时间序列异常检测方法进行对比实验，对比实验结果如表 2 所示. 通过对 SELAD 在 4 个数据集上的 *F*1 分数、AUC-ROC 以及 AUC-PR 取平均，SELAD 方法的平均 *F*1 分数为 0.915 3，相比所有对比算法的 *F*1 分数提高了 17.87~90.77 个百分点；其平均 AUC-ROC 为 0.990 1，相比所有对比方法的 AUC-ROC 提高了 12.69%~32.98%；其平均 AUC-PR 为 0.891 8，相比

Table 2 Comparative Results of Anomaly Detection

表 2 异常检测对比结果

方法	ASD			SMD		
	*F*1	AUC-ROC	AUC-PR	*F*1	AUC-ROC	AUC-PR
COPOD	0.558 4(0.448 1)	0.872 6(0.123 5)	0.476 4(0.626 8)	0.833 2(0.085 0)	**0.989 4**(−0.000 3)	0.840 5(0.051 0)
iForest	0.705 9(0.145 5)	0.927 5(0.057 0)	0.651 2(0.190 1)	0.865 4(0.044 6)	0.985 5(0.003 7)	0.863 5(0.023 0)
DSVDD	0.757 3(0.067 7)	0.808 5(0.212 6)	0.767 2(0.010 2)	0.837 9(0.078 9)	0.836 7(0.182 1)	0.866 8(0.019 2)
LSTM-VAE	0.501 4(0.612 7)	0.859 2(0.141 1)	0.396 1(0.956 6)	0.820 2(0.102 2)	0.927 5(0.066 4)	0.780 4(0.132 0)
USAD	0.210 2(2.846 8)	0.598 8(0.637 3)	0.128 7(5.021 8)	0.566 2(0.596 6)	0.816 5(0.211 4)	0.472 2(0.870 8)
TS-CP2	0.302 7(1.671 3)	0.759 7(0.290 5)	0.188 3(3.115 8)	0.400 1(1.259 4)	0.745 1(0.327 5)	0.324 0(1.726 5)
GPT4TS	0.517 7(0.561 9)	0.531 0(0.846 3)	0.662 0(0.170 7)	0.711 8(0.270 0)	0.739 7(0.337 2)	0.759 5(0.163 1)
SELAD（本文）	**0.808 6**	**0.980 4**	**0.775 0**	**0.904 0**	0.989 1	**0.883 4**
平均增长率	0.907 7	0.329 8	1.441 7	0.348 1	0.161 1	0.426 5
方法	MSL			SMAP		
	*F*1	AUC-ROC	AUC-PR	*F*1	AUC-ROC	AUC-PR
COPOD	0.923 6(0.050 7)	0.983 0(0.012 0)	0.921 2(0.023 9)	0.815 4(0.199 7)	0.967 4(0.029 7)	0.769 6(0.254 4)
iForest	0.842 7(0.151 5)	0.976 4(0.018 8)	0.807 4(0.168 2)	0.746 4(0.310 6)	0.911 1(0.093 3)	0.693 5(0.392 1)
DSVDD	0.888 0(0.092 8)	0.851 6(0.168 2)	**0.989 1**(−0.046 4)	0.794 5(0.231 2)	0.743 4(0.339 9)	**0.992 6**(−0.027 4)
LSTM-VAE	0.875 2(0.108 8)	0.975 6(0.019 7)	0.839 2(0.123 9)	0.879 6(0.112 1)	0.983 8(0.012 5)	0.846 3(0.140 7)
USAD	0.775 7(0.251 0)	0.900 6(0.104 6)	0.716 8(0.315 8)	0.647 2(0.511 4)	0.881 0(0.130 6)	0.586 0(0.647 4)
TS-CP2	0.780 2(0.243 8)	0.938 8(0.059 7)	0.736 3(0.281 0)	0.633 8(0.543 4)	0.794 2(0.254 2)	0.596 7(0.617 9)
GPT4TS	0.717 5(0.352 5)	0.660 7(0.505 7)	0.977 3(−0.034 9)	0.678 3(0.442 1)	0.610 4(0.631 9)	0.985 8(−0.020 7)
SELAD（本文）	**0.970 4**	**0.994 8**	0.943 2	**0.978 2**	**0.996 1**	0.965 4
平均增长率	0.178 7	0.126 9	0.118 8	0.335 8	0.213 2	0.286 4

注：括号中的数据为 SELAD 方法相比于对比方法的增长率. 平均增长率是 SELAD 与所有对比方法的平均提升程度. 黑体数值表示最优结果.

所有对比方法的 AUC-PR 提高了 11.88%~144.17%. SELAD 方法在 4 个数据集的 *F*1 分数高于 7 种多维时间序列异常检测方法，说明保留多维时间序列数据完整周期性的多窗口划分集成学习方法可以帮助模型更好地学习多维时序数据的正常模式.

具体来说，ASD 数据集是一个复杂的多维智能运维数据集，不同维度的时间周期性和数据特征并不一致，因此采用 SELAD 方法可以更好地保留不同维度的周期性特征，从而更好地检测 ASD 数据集中的各种异常. 在 ASD 数据集上，SELAD 方法的 *F*1 分数相比对比方法中的 *F*1 分数平均提高了 90.77 个百分点，AUC-ROC 指标相较于对比方法中的 AUC-ROC 指标平均提高了 32.98%，AUC-PR 指标相较于对比方法中的 AUC-PR 指标平均提高了 144.17%.

相比于智能运维数据集 ASD 中各个参数耦合的复杂异常，传感器数据集 MSL 和 SMAP 中的异常多为点异常和上下文异常. 在 MSL 数据集上 SELAD 方法的 *F*1 分数相比对比方法的 *F*1 分数平均提高了 17.87 个百分点，AUC-ROC 指标相较于对比方法中的 AUC-ROC 指标平均提高了 12.69%，AUC-PR 指标相较于对比方法中的 AUC-PR 指标平均提高了 11.88%. SELAD 方法在 MSL 数据集上的 *F*1 分数相较于对比算法中的 *F*1 分数平均提高了 33.58 个百分点，AUC-ROC 指标相较于对比方法中的 AUC-ROC 指标平均提高了 21.32%，AUC-PR 指标相较于对比方法中的 AUC-PR 指标平均提高了 28.64%. 说明在传感器多维时间序列数据上，SELAD 方法可以通过多窗口划分机制更好地保留多维时序数据中的上下文异常.

SMD 数据集是大规模的多维服务器群组数据集. 对于大规模数据集，SELAD 方法也可以得到较好的异常测试 *F*1 分数，相比对比方法中的 *F*1 分数平均提高了 34.81 个百分点，AUC-ROC 指标相较于对比方法中的 AUC-ROC 指标平均提高了 16.11%，AUC-PR 指标相较于对比方法中的 AUC-PR 指标平均提高了 42.65%. 说明在大规模多维服务器群组数据集上，SELAD 方法可以避免维度融合带来的过拟合问题.

综上，SELAD 方法可以更加有效地对数据的每一维进行完整周期特性的多窗口划分，通过融合大

模型和 LSTM 模型的混合专家模型对输入数据进行正常时序模式的学习，从而可以更好地构建准确的特征表示，弥补了目前的多维时间序列异常检测方法中采用统一划分无法较好地保留多维时序数据每一维周期性特征的缺陷.

3.6 消融分析

本节在 4 个实验数据集上对 SELAD 方法进行了消融实验，进一步验证本文设计的多维时序窗口划分模块和混合专家模块的合理性和有效性.

1）SELAD 模型. 使用大模型 GPT-2 并且采用多窗口划分机制进行窗口划分.

2）SELAD w/o（without）GPT 模型. 在 SELAD 原始模型中的集成学习框架中，去除了 GPT-2 模型，使用 LSTM 模型进行不同划分子数据的学习，依旧保留多窗口划分机制.

3）SELAD w/o（without）WS 模型. 在 SELAD 原始模型中，去除多窗口划分机制，采用参数化划分窗口大小. 同时，保留了大模型 GPT-2 和 LSTAM 模型的集成学习框架.

4）SELAD w/o（without）ALL 模型. 在 SELAD 原始模型中同时去除大模型和多窗口划分机制. 此时，模型退化为采用参数化划分窗口的 LSTM 模型.

如表 3 所示，在所有的实验数据集上，加入了多窗口划分机制的 SELAD w/o GPT 模型相较于普通的采用参数化划分窗口的 LSTM 模型的 SELAD w/o ALL 模型的平均 $F1$ 分数提高了 6.77 个百分点，说明多窗口划分机制的引入有助于更好地保留多维时间序列数据每一维的周期性特征. 在 4 个数据集上，加入了混合专家模块的 SELAD w/o WS 模型相比传统的 LSTM 模型的 SELAD w/o ALL 模型的平均 $F1$ 分数提高了 8.02 个百分点，说明大模型对多维时间序列数据正常模式的学习能力更强.

Table 3 Ablation Study

表 3 消融实验

模型	ASD			SMD		
	$F1$	AUC-ROC	AUC-PR	$F1$	AUC-ROC	AUC-PR
SELAD(本文)	**0.808 6**	**0.980 4**	**0.775 0**	**0.9040**	**0.989 1**	**0.883 4**
SELAD w/o GPT	0.782 8(0.032 9)	0.944 1(0.038 4)	0.739 5(0.048 0)	0.808 6(0.117 9)	0.987 3(0.001 8)	0.753 8(0.171 9)
SELAD w/o WS	0.785 1(0.029 9)	0.934 3(0.049 3)	0.742 5(0.043 7)	0.848 2(0.065 7)	0.989 1(0)	0.804 7(0.097 8)
SELAD w/o ALL	0.708 6(0.141 1)	0.916 4(0.069 8)	0.648 3(0.195 4)	0.770 5(0.173 3)	0.985 2(0.003 9)	0.714 8(0.235 8)
平均增长率	6.8	5.25	9.57	0.119	0.001 9	0.168 5
模型	MSL			SMAP		
	$F1$	AUC-ROC	AUC-PR	$F1$	AUC-ROC	AUC-PR
SELAD(本文)	**0.970 4**	**0.994 8**	**0.943 2**	**0.978 2**	**0.996 1**	**0.965 4**
SELAD w/o GPT	0.955 5(0.015 6)	0.992 5(0.002 3)	0.939 2(0.004 2)	0.909 6(0.075 4)	0.987 2(0.009 0)	0.858 9(0.124)
SELAD w/o WS	0.952 8(0.018 4)	0.992 6(0.002 2)	0.909 1(0.037 5)	0.910 9(0.073 8)	0.990 9(0.005 2)	0.868 6(0.111)
SELAD w/o ALL	0.891 0(0.089 1)	0.963 9(0.032 0)	0.873 0(0.080 4)	0.867 3(0.128 0)	0.989 4(0.006 7)	0.820 2(0.177)
平均增长率	0.041 1	0.012 2	0.040 7	0.092 3	0.007 0	0.137 5

注：括号中的数据为 SELAD 模型相比于其他模型的增长率. 平均增长率是 SELAD 与所有对比方法的平均提升程度. 黑体数值表示最优结果.

在 4 个数据集上，SELAD 模型相比只保留 GPT 模型的 SELAD w/o WS 模型的 $F1$ 分数提升了 1.84~7.38 个百分点；相比加入了多窗口划分机制的 SELAD w/o GPT 模型的 $F1$ 分数提升了 1.56~11.79 个百分点；相比 SELADw/o ALL 模型的 $F1$ 分数提升了 4.11~11.9 个百分点.

消融实验结果分析表明，多维时序窗口划分模块可以更好地保留多维时间序列数据中每一维的周期特性，同时，融合大模型 GPT 和 LSTM 模型的混合专家模块可以避免传统模型对长时间序列数据的记忆瓶颈问题和大模型的训练时间过长的问题，使得模型可以更好地学习多维时间序列数据的时间序列的正常模式，从而表现出更好的异常检测性能.

3.7 时间损耗分析

大模型时代下，超大规模的数据集会提高模型的训练资源开销以及训练推理时间，使其在现实问题中不具有可用性. 为了验证 SELAD 方法在智算运维多维时间序列异常检测场景中的可用性，本文在 ASD 智能运维数据集上对 SELAD 方法和 3 种时间序列异常检测方法进行单轮训练时间和推理时间对

比评估. 实验结果如图 2 表示, 对比 3 种多维时间序列异常检测方法, 从单轮训练时间来看, SELAD 方法优于 GPT4TS 方法和 USAD 方法, 逊于 LSTM-AE 方法. 在智算运维中的异常检测场景中, SELAD 方法的训练时间开销符合需求, 同时, 智算运维更加关注异常检测方法的实时性. 因此, 在智算运维异常检测领域, 异常检测方法的推理时间比训练时间更加关键. 从方法推理时间来看, SELAD 方法优于 GPT4TS 方法和 LSTM-AE 方法, 逊于 USAD 方法. 这得益于 SELAD 方法采用了周期性的窗口划分策略, 考虑到多维时间序列中不同维度周期性的影响, 从更小的局部窗口中学习数据的上下文信息, 在避免了信息截断的同时可以降低模型的训练和推理时间.

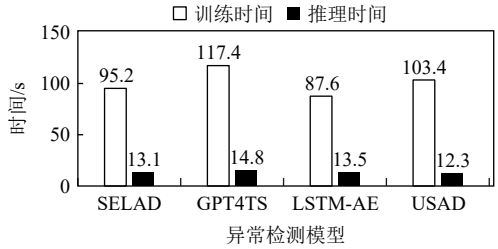


Fig. 2 Graph of time loss

图 2 时间损耗图

4 结 论

多维时间序列异常检测是算力网络智算运维中的关键任务. 本文提出了一种基于多窗口划分的无监督多维时间序列异常检测方法 SELAD, 该方法通过对多维时间序列数据进行非参数化的窗口划分, 通过融合大模型和 LSTM 模型的混合专家模型学习输入数据的正常模式, 构建更准确的特征表示, 以重构误差定义损失函数进行多维时间序列异常检测. 通过在 4 个多维时间序列数据集的对比实验, SELAD 的异常检测性能优于目前先进的异常检测方法, 通过消融实验验证了本文中采用多窗口划分机制可以保留多维时间序列数据每一维的特征, 同时混合专家系统的框架设计可以更好地学习每一维的时序正常模式, 从而通过重构误差进行多维时间序列的异常检测.

在未来的研究中, 我们计划进一步优化算法效率, 并探索更鲁棒的周期性特征提取方法. 优化混合专家模块的网络结构, 尝试通过设计更符合时间序列特征的门控网络来增强对大模型和传统模型的集成效果, 提高多维时间序列异常检测效果.

作者贡献声明: 王泽南提出主要研究思路, 完成实验并撰写论文; 王意洁设计实验方案, 提供指导意见, 并负责修改和审核论文; 周小晖协助完成部分实验, 并提出论文修改建议; 熊旭东参与论文的修改工作.

参 考 文 献

[1] Narayanan D, Shoeybi M, Casper J, et al. Efficient large-scale language model training on GPU clusters using Megatron-LM[C/OL]// Proc of the 33rd Int Conf for High Performance Computing, Networking, Storage and Analysis. Piscataway, NJ: IEEE, 2021[2024-10-10]. https://ieeexplore.ieee.org/document/9910057

[2] Wang Qinlong, Sang Bo, Zhang Haitao, et al. DLRover: An elastic deep training extension with auto job resource recommendation[J]. arXiv preprint, arXiv: 2304.01468, 2023

[3] Zhou Xiaohui, Wang Yijie, Xu Hongzuo, et al. Fusion learning based unsupervised anomaly detection for multi-dimensional time series[J]. Journal of Computer Research and Development, 2023, 60(3): 496−508 (in Chinese)
(周小晖, 王意洁, 徐鸿祚, 等. 基于融合学习的无监督多维时间序列异常检测[J], 计算机研究与发展, 2023, 60(3): 496−508)

[4] Blázquez-García A, Conde A, Mori U, et al. A Review on outlier/anomaly detection in time series data[J]. ACM Computing Surveys, 2022, 54(3): 1−33

[5] Hochreiter S, Schmidhuber J. Long short-term memory[J]. Neural Computation, 1997, 9(8): 1735−1780

[6] 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

[7] Audibert J, Michiardi P, Guyard F, et al. USAD: Unsupervised anomaly detection on multivariate time series[C]//Proc of the 26th ACM SIGKDD Int Conf on Knowledge Discovery & Data Mining. New York: ACM, 2020: 3395−3404

[8] Zhang Yuxin, Wang Jindong, Chen Yiqiang, et al. Adaptive memory networks with self-supervised learning for unsupervised anomaly detection[J]. IEEE Transactions on Knowledge and Data Engineering, 2022, 35(12): 12068−12080

[9] Campos D, Kieu T, Guo Chenjuan, et al. Unsupervised time series outlier detection with diversity-driven convolutional ensembles[J]. arXiv preprint, arXiv2111.11108, 2021

[10] Su Ya, Zhao Youjian, Niu Chenhao, et al. Robust anomaly detection for multivariate time series through stochastic recurrent neural network[C]//Proc of the 25th ACM SIGKDD Int Conf on Knowledge Discovery & Data Mining. New York: ACM, 2019: 2828−2837

[11] Hundman K, Constantinou V, Laporte C, et al. Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding[C]//Proc of the 24th ACM SIGKDD Int Conf on Knowledge Discovery & Data Mining. New York: ACM, 2018: 387−395

[12] Wu Wentai, He Ligang, Lin Weiwei, et al. Developing an unsupervised real-time anomaly detection scheme for time series with

multi-seasonality[J]. IEEE Transactions on Knowledge and Data Engineering, 2020, 34(9): 4147−4160

[13] Deng Ailin, Hooi B. Graph neural network-based anomaly detection in multivariate time series[C]//Proc of the 35th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2021: 4027−4035

[14] Jin Ming, Zhang Yifan, Chen Wei, et al. Position: What can large language models tell us about time series analysis[J]. arXiv preprint, arXiv: 2402.02713, 2024

[15] Liu Jun, Zhang Chaoyun, Qian Jiaxu, et al. Large language models can deliver accurate and interpretable time series anomaly detection[J]. arXiv preprint, arXiv: 2405.15370, 2024

[16] Dang Weixia, Zhou Biyu, Zhang Weigang, et al. Time series anomaly detection based on language model[C]//Proc of the 11th ACM Int Conf on Future Energy Systems(e-Energy'20). New York: ACM, 2020: 544−547

[17] Chen Song, Liao Hai. BERT-Log: Anomaly detection for system logs based on pre-trained language model[J]. Applied Artificial Intelligence, 2022, 36(1): 2145642

[18] Zhou Tian, Niu Peisong, Wang Xue, et al. One fits all: Power general time series analysis by pretrained LM[J]. arXiv preprint, arXiv: 2302.11939, 2023

[19] Sarda K, Namrud Z, Rouf R, et al. ADARMA auto-detection and auto-remediation of microservice anomalies by leveraging large language models[C]//Proc of the 33rd Annual Int Conf on Computer Science and Software Engineering. New York: ACM, 2023: 200−205

[20] Khlaisamniang P, Khomduean P, Saetan K, et al. Generative AI for self-healing systems[C/OL]//Proc of the 18th Int Joint Symp on Artificial Intelligence and Natural Language Processing. Piscataway, NJ: IEEE, 2023[2025-02-20]. https://ieeexplore.ieee.org/document/10354608

[21] Garza A, Challu C, Mergenthaler-Canseco M. TimeGPT-1[J]. arXiv preprint, arXiv: 2310.03589, 2023

[22] Goswami M, Szafer K, Choudhry A, et al. MOMENT: A family of open time-series foundation models[J]. arXiv preprint, arXiv: 2402.03885, 2024

[23] Ansari A F, Stella L, Turkmen C, et al. Chronos: Learning the language of time series[J]. arXiv preprint, arXiv: 2403.07815, 2024

[24] Li Longyuan, Yan Junchi, Wen Qingsong, et al. Learning robust deep state space for unsupervised anomaly detection in contaminated time-series[J]. IEEE Transactions on Knowledge and Data Engineering, 2023, 35(6): 6058−6072

[25] Liu F T, Ting Kaiming, Zhou Zhihua. Isolation forest[C/OL]//Proc of the 8th IEEE Int Conf on Data Mining. Piscataway, NJ: IEEE, 2008[2025-02-20]. https://ieeexplore. ieee.org/document/4781136

[26] Li Zheng, Zhao Yue, Botta N, et al. COPOD: Copula-based outlier detection[C]//Proc of the 20th IEEE Int Conf on Data Mining. Piscataway, NJ: IEEE, 2020: 1118−1123

[27] Ruff L, Görnitz N, Deecke L, et al. Deep one-class classification[C/OL]//Proc of the 35th Int Conf on Machine Learning. 2018[2025-02-20]. https://proceedings.mlr.press/v80/ruff18a/ruff18a.pdf

[28] Deldari S, Smith D V, Xue H, et al. Time series change point detection with self-supervised contrastive predictive coding[C]//Proc of the Int World Wide Web Conf Committee. New York: ACM, 2021: 3124−3135



Wang Zenan, born in 2001. Master candidate. His main research interests include anomaly detection, deep learning, and ensemble learning.

王泽南, 2001 年生. 硕士研究生. 主要研究方向为异常检测、深度学习、集成学习.



Wang Yijie, born in 1971. PhD, professor, PhD supervisor. Her main research interests include big data analysis, cloud computing, and distributed storage.

王意洁, 1971 年生. 博士, 教授, 博士生导师. 主要研究方向为大数据分析、云计算、分布式存储.



Zhou Xiaohui, born in 2000. Master candidate. His main research interests include AIOps, anomaly detection, and deep learning.

周小晖, 2000 年生. 硕士研究生. 主要研究方向为智能运维、异常检测、深度学习.



Xiong Xudong, born in 2003. Bachelor. His main research interests include time series anomaly detection and deep learning.

熊旭东, 2003 年生. 学士. 主要研究方向为时序数据异常检测、深度学习.