计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2024,33(3):24−33 [doi: 10.15888/j.cnki.csa.009423]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于图偏差网络的外部自编码器时间序列异常检测①

张孚容, 顾　磊

(南京邮电大学 计算机学院、软件学院、网络空间安全学院, 南京 210023)
通信作者: 张孚容, E-mail: 1285338667@qq.com

摘　要: 随着互联网和连接技术的提高, 传感器产生的数据逐渐趋于复杂化. 深度学习方法在处理高维数据的异常检测方面取得较好的进展, 图偏差网络 (graph deviation network, GDN) 学习传感器节点之间关系来预测异常, 并取得一定的效果. 针对图偏差网络模型缺少对时间依赖性以及异常数据不稳定的处理, 提出了基于图偏差网络的外部自编码器模型 (graph deviation network-based external attention autoencoder, AEEA-GDN) 深度提取表征, 此外在模型训练时引入自适应学习机制, 帮助网络更好地适应异常数据的变化. 在 3 个现实收集传感器数据集上的实验结果表明, 基于图偏差网络的外部自编码器模型比基线方法更准确地检测异常, 且总体性能更优.

关键词: 异常检测; 图偏差网络; 自编码器; 外部注意力机制; 自适应学习

引用格式: 张孚容,顾磊.基于图偏差网络的外部自编码器时间序列异常检测.计算机系统应用,2024,33(3):24–33. http://www.c-s-a.org.cn/1003-3254/9423.html

Time Series Anomaly Detection With External Autoencoder Based on Graph Deviation Network

ZHANG Fu-Rong, GU Lei

(School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China)

Abstract: With the improvement of the Internet and connection technology, the data generated by sensors is gradually becoming complex. Deep learning methods have made great progress in anomaly detection of high-dimensional data. The graph deviation network (GDN) learns the relationship between sensor nodes to predict anomalies and has achieved certain results. Since the GDN model fails to deal with time dependence and instability of abnormal data, an external attention autoencoder based on GDN (AEEA-GDN) is proposed to deeply extract features. In addition, an adaptive learning mechanism is introduced during model training to help the network better adapt to changes in abnormal data. Experimental results on three real-world collected sensor datasets show that the AEEA-GDN model can more accurately detect anomalies than baseline methods and has better overall performance.

Key words: anomaly detection; graph deviation network (GDN); autoencoder; external attention mechanism; adaptive learning

异常检测 (anomaly detection)[1], 又称离群点检测 (outlier detection), 是检测不匹配预期模式或与大多数数据实例显著不同的数据点, 这些被检测出的数据点被称为异常点.

随着物联网和传感器数据在物理系统 (cyber physical systems, CPS) 中的快速增长, 例如工业系统和数据中心, 需要监视这些设备以防止受到攻击. 异常检测在计算机视觉[2]、数据挖掘[3]、自然语言处理等实

① 基金项目: 国家自然科学基金 (61972210)
收稿时间: 2023-09-02; 修改时间: 2023-10-08; 采用时间: 2023-10-20; csa 在线出版时间: 2023-12-25
CNKI 网络首发时间: 2023-12-27

际应用领域发挥着越来越重要的作用, 已成为许多研究人员和从业者感兴趣的领域, 现在是数据挖掘和质量保证的主要任务之一[4].

在该领域中, 异常检测技术可分为基于线性模型的方法[5]、基于距离的方法[6]、支持向量机的方法[7]、基于密度的方法[8]. 然而, 这些方法通常以相对简单的方式建模传感器数据之间的相互关系. 例如, 仅捕获线性关系, 这对于许多现实世界的复杂、高度非线性关系是不足够的.

近年来, 基于深度学习的技术已经使得在高维数据集中进行异常检测变得更加容易, 相较于其他方法其性能有较大提升. 例如, 自编码器[9]是一种流行的用于异常检测的方法, 它基于反向传播算法与最优化算法, 将输入数据在神经网络中进行特征学习映射, 从而得到一个重构的输出数据. 深度自编码高斯混合模型 (deep autoencoding Gaussian mixture model for unsupervised anomaly detection, DAGMM)[10]可以处理高维数据、非线性数据以及自动学习数据的分布. 无监督异常检测 (unsupervised anomaly detection, UASD)[11]的组成包括基于自编码方法和基于生成对抗网络 (generative adversarial network, GAN)[12]的方法. 基于 LSTM (long short-term memory)[13]的方法在公开的数据集上测试均表现出良好性能. 但是, 大多数方法忽略了传感器数据彼此之间的相关性.

利用多元时间序列中传感器数据之间的复杂关系学习特征来提高异常检测的性能. 图神经网络 (graph neural network, GNN)[14]利用图结构数据来进行异常检测取得了良好的成效. 图卷积网络 (graph convolutional network, GCN)[15]是图神经网络和卷积神经网络结合的异常检测模型, 卷积核在图上进行局部连接和信息传播, 有效地利用图结构的拓扑信息和节点信息, 提高图数据的表征能力且具有较强的泛化能力. 图注意力网络 (graph attention network, GAT)[16]是一种基于注意力机制的图神经网络, 可以动态地计算每个节点对其邻居节点的注意力权重, 从而捕捉节点之间的非均匀性. 但是, 这一类检测模型在时间序列异常检测过程中并没有考虑不同的传感器具有非常不同的行为. 例如, 一个传感器可能测量水压, 而另一个传感器可能测流量, 典型的 GNN 使用相同的模型参数来模拟每个节点的行为. 图偏差网络[17]能够学习到不同传感器之间的不同行为, 从而捕捉传感器数据之间的依赖性和相互作用, 并且再将数据送入到图注意力模型学习, 能够提供异常检测的可解释性. 但是图偏差网络缺少对时间依赖性的学习以及对异常数据的不稳定性的处理.

综上所述, 本文提出了一种基于图偏差网络的外部自编码器时间序列异常检测模型, 该模型将时间序列数据分别送入由外部注意力机制、自动编码器组成的时间依赖性学习模块学习时间依赖性以及送入图结构学习模块学习传感器数据依赖性, 经过不同角度的特征提取, 再送入图预测网络进行训练预测, 同时考虑到时间异常数据可能会随着时间的改变而发生推移, 引入自适应学习机制帮助网络更好地适应异常数据的变化, 从而提高时间序列异常检测的准确率. 本文的主要贡献如下: (1) 提出了基于图偏差网络的外部自编码器时间序列异常检测方法, 将时间依赖性特征和传感器依赖性特征结合学习; (2) 对于时间异常数据的不稳定性, 采用自适应学习机制提高网络的准确性; (3) 对 3 个数据集进行了实验, 本文提出的模型相较于其他异常检测模型拥有更好的结果, 均提升了时间序列异常检测准确率.

1 相关模型和技术

1.1 图偏差网络 GDN 模型

2005 年, Gori 等人[18]首次提出图神经网络 GNN, 这种网络是基于深度学习的, 旨在直接对图结构数据进行学习, 以揭示图中节点和边的内在规律以及更深层次的语义特征. 该模型的主要理念是通过学习一个映射函数来实现这一目标, 通过该映射图中的节点可以聚合它自己的特征与它的邻居特征生成节点的新表示. 一般来说, GNN 假设一个节点的状态受到其邻居节点状态的影响. 基于 CNN 的图卷积网络 GCN 通过聚合其一步邻居的表示来建模一阶节点的特征表示. 相关的变体在时间依赖的问题中表现出成功, 例如, 基于 GNN 的模型可以在交通预测[19]任务中表现良好. 在推荐系统[20]和相关应用[21]中验证了 GNN 建模大规模多关系数据的有效性. 然而, 这些方法使用相同的模型参数来建模每个节点的行为, 在表示不同传感器不同的行为时面临限制. 此外, GNN 通常需要图结构作为输入, 而在模型的设置中, 图结构最初是未知的, 需要从数据中学习.

2021 年, Deng 等人[17]提出的图偏差网络 GDN 是一种基于图神经网络的多元时间序列异常检测方法.

该模型包括 4 个部分: 传感器数据编码模块、图结构学习模块、基于图注意力预测模块、计算图偏差分数模块. 算法的具体步骤: 首先对每个传感器获取到数据进行初始化, 即图 1 中传感器数据编码模块, 可用来在图结构学习中判断传感器数据之间的相关性; 该模型考虑除了自身与其他传感器均存在依赖性, 经过图结构的学习得到该节点与其他节点的依赖性数据, 归一化后, 选择这样的归一化点积, 以此组成邻接矩阵并认为该传感器与其他传感器存在较强的关联性, 并用邻接矩阵来存储邻接关系; 将得到的邻接关系和初始化的数据结合送入图注意力预测网络预测某时刻的传感器数据的值; 最后利用观测值和预测值计算得到异常分数, 并和阈值进行比较, 判断异常. 该模型的具体流程如图 1 所示.

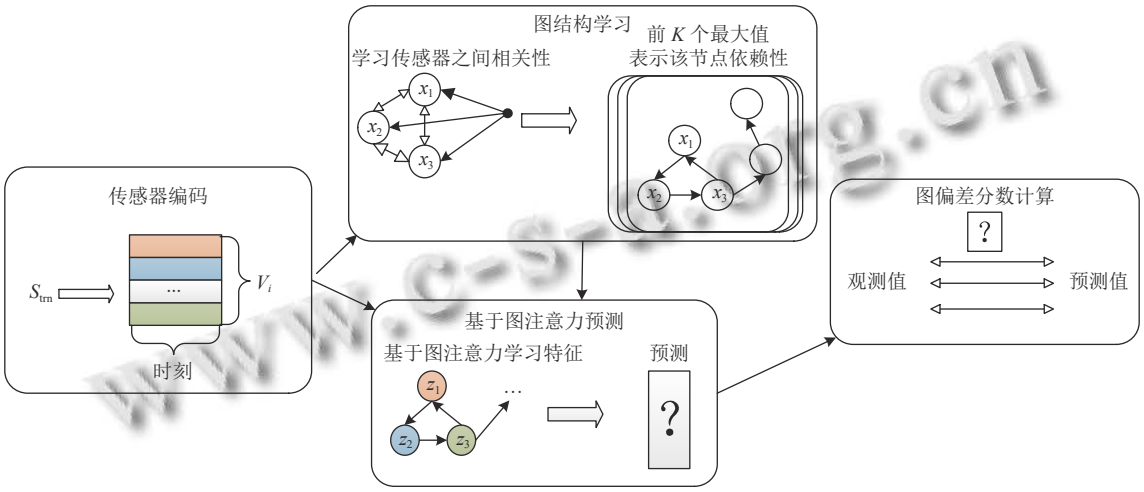

图 1 图偏差网络模型

1.2 自编码器

1986 年, Rumelhart 等人[22]提出自编码器, 它利用输入数据本身作为监督, 来学习一种数据的压缩表示, 被称为神经网络的预训练方法. 该方法依赖于反向传播算法和优化技巧 (如梯度下降法), 利用输入数据本身作为监督信号, 引导神经网络试图学习一种映射关系, 从而实现重构输出的生成. 在时间序列异常检测场景下, 异常对于正常来说是少数, 使用自编码器重构出来的输出跟原始输入的差异超出一定阈值的话, 原始时间序列即存在了异常.

算法模型由两个核心组成部分组成: 编码器 (encoder) 和解码器 (decoder). 编码器旨在将高维输入转化为低维隐变量, 从而引导神经网络学习最富信息的特征; 解码器旨在把隐藏层的隐变量还原到初始维度, 最好的状态就是解码器的输出能够完美地或者近似恢复出原来的输入. 考虑到时间序列异常检测的时效性, 本文采用全连接层来架构自编码器. 在图 2 中以红色虚线框表示.

1.3 外部注意力机制

2017 年, Vaswani 等人[23]提出自注意力机制, 它能够捕捉长距离依赖关系, 有助于提高各种自然语言处理[24]和计算机视觉[25]任务的性能. 该算法的原理是: 对于每个输入向量, 都计算它与其他输入向量的点积 (或者加上一个可学习的权重矩阵), 然后通过一个 Softmax 激励函数得到一个概率分布, 这个概率分布表示每个输入向量对当前输出向量的贡献程度. 然后, 将这个概率分布与输入向量相乘, 得到输出向量. 但是这导致了样本中位置数量的二次计算复杂度. 此外, 自注意力集中在单个样本内不同位置之间的自相关性上, 而忽略了与其他样本的潜在相关性, 并且只考虑数据样本中元素之间的关系, 而忽略了不同样本中元素间的潜在关系, 这可能会限制自注意的能力和灵活性.

2022 年, Guo 等人[26]提出的外部注意力机制, 建立在一对独立的、可训练共享内存单元之上, 通过两个级联的线性层和两个归一化层的简单组合, 就可以落实. 外部注意力机制的优点是, 它可以减少计算复杂度, 并且也可以捕捉不同样本之间的联系, 而不是只关注同一个样本内部的相关性. 在图 2 中以绿色虚线框表示.

1.4 自适应学习

自适应学习是深度学习中的一种技术, 可以自动调整学习率和其他超参数, 并减少手动调整超参数的工作量. 在深度学习中, 学习率是一个非常重要的超参数, 它控制着模型在每次迭代中更新权重的速度. 如果

学习率太小, 模型将需要更多的迭代才能收敛; 如果学习率太大, 模型可能会在训练过程中发生震荡或不收敛.

在本文代码中, 使用了 Adam 优化算法和 ReduceLROnPlateau 调度器来自动调整学习率. ReduceLROnPlateau 调度器会监控验证损失, 并在损失停止下降时降低学习率, 因子参数决定要降低多少学习率, 耐心参数决定在降低学习率之前等待多少个周期. 这种方法可以自动调整学习率, 以提高模型的性能. 在图 2 中以橘色虚线框表示.

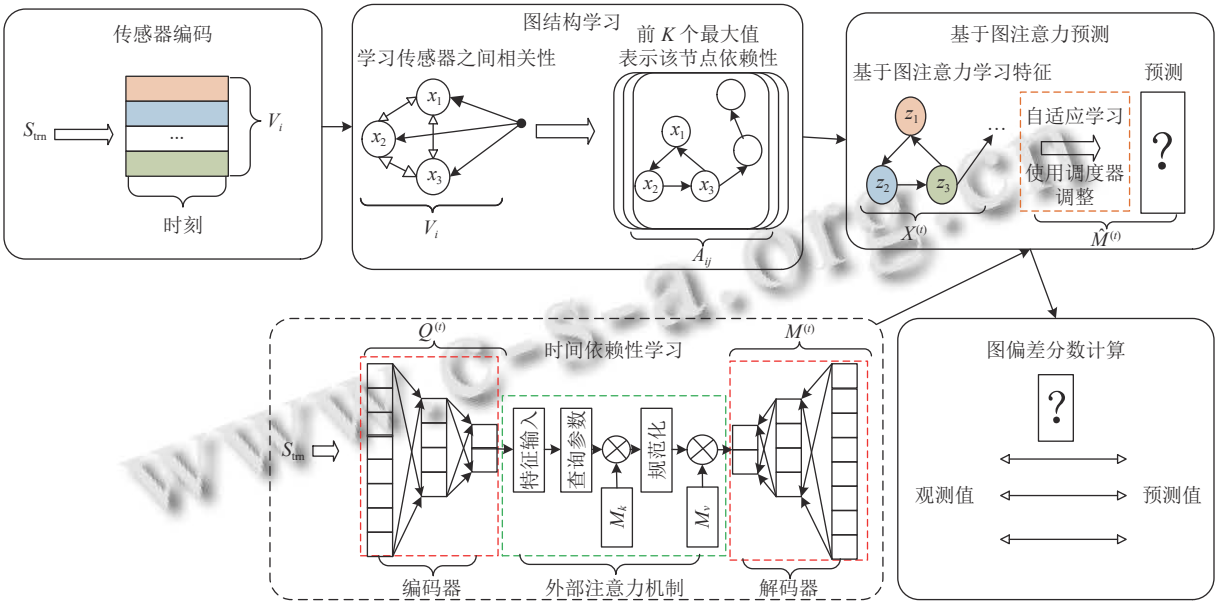

图 2 图偏差网络的外部自编码器异常检测模型

2 AEEA-GDN 异常检测模型

在本文中, 训练数据来自N个传感器数据在T_{trn}范围数据$S_{\text{trn}}=[S_{\text{trn}}^{(1)},\cdots,S_{\text{trn}}^{(T_{\text{trn}})}]$, 在每个时刻$t$的数据$S_{\text{trn}}^{(t)}\in R^N$ 形成一个 N 维向量, 表示N个传感器数据对应的值, $t\in[1,T_{\text{trn}}]$. 由于采用无监督训练模型, 训练数据中一般仅包含正常数据, 模型的目标是检测测试数据中的异常, 这些数据来自相同的N个传感器, 测试数据集表示为$S_{\text{tst}}=[S_{\text{tst}}^{(1)},\cdots,S_{\text{tst}}^{(T_{\text{tst}})}]$. 其中, T_{trn}表示训练数据时刻t范围, S_{trn}是按照采样频率形成的训练数据集, T_{tst}表示测试数据时刻t范围, S_{tst}是按照采样频率形成的测试数据集. 异常检测是一个二分类任务, 所以输出是测试数据集在T_{tst}个时刻中的二进制标签, 表示每个测试时刻t是否为异常, 即$a(t)\in\{0,1\}$, 其中$a(t)=1$表示数据在时刻t出现异常.

2.1 模型结构

GDN 网络在没有先验信息的情况下, 学习某个传感器除了自身之外与其他传感器之间的关系, 得到不同传感器之间的不同行为, 从而捕捉传感器之间的依赖性和相互作用, 并且将学习特征送入到图注意力模型学习, 能够提供异常检测的可解释性. 但是, 该模型未考虑在同一时刻下, 各个传感器呈现的特征以及异常点的特征, 所以引入时间依赖性学习模块来学习检测异常. 例如, 如果一个时间序列中的时间数据点之间存在某种周期性关系, 则可以使用这种关系来检测异常. 此外, 在现实环境中, 异常数据是罕见的, 会受到时间变化的影响, 使用同一参数训练模型是不合理的, 为此引入自适应学习监控验证损失, 并在损失停止下降时降低学习率, 因子参数决定要降低多少学习率, 耐心参数决定在降低学习率之前等待多少个周期, 这种方法可以自动调整学习率, 以提高模型的性能.

基于图偏差网络的外部自编码器模型结构如图 2 所示. 该模型由 5 个部分组成: 传感器数据编码模块、图结构学习模块、时间依赖性学习模块、基于图自适应注意力预测模块、图偏差分数计算模块, 其中虚线部分属于本文的创新点模块, 实线部分属于 GDN 模块.

2.2 传感器数据编码模块

在工业环境中, 不同的传感器可能具有不同的特性, 并且这些特性可以以复杂的方式关联. 例如, 假设在两碗水中放置了检测水质和水温的传感器, 那么, 这两个水质、水温传感器的行为可能相似. 但是, 同样有

可能的是, 同一碗水中的传感器会表现出很强的相关性. 因此, 本文为了灵活表示每个传感器, 采用多维的方式描述每个传感器的特性.

在训练数据集S_{trn}中, $S_{\text{trn}}^{(t)}$表示所有传感器在时刻t的值, 其中每个传感器特征的嵌入向量用式 (1) 表示:

$$V_i \in R^d,\ i \in \{1,2,\cdots,N\} \tag{1}$$

其中, V_i表示第i个传感器在连续的d个时刻内数值所组成的向量, V_i相似的传感器可能存在较强的相似关系.

2.3 图结构学习模块

因为每个传感器之间的相似关系不同, 也可能两个传感器之间不存在相似关系, 所以本文采用有向图来表示传感器之间的不对称依赖关系, 考虑到传感器之间的依赖关系存在先验信息的情况下, 本文假设除了自身节点与其他传感器节点均存在依赖关系, 通过式 (2) 表示传感器之间的候选关系:

$$C_i \subseteq \{1,2,\cdots,N\}\backslash\{i\} \tag{2}$$

其中, N表示有N个传感器, C_i表示第i个传感器与其他传感器之间的候选关系.

通过式 (3) 计算得到传感器i与传感器j之间的相似度:

$$e_{ji} = \frac{V_i^{\mathrm{T}} V_j}{\|V_i\| \cdot \|V_j\|} \tag{3}$$

其中, $j \in C_i$, e_{ji}表示传感器i与传感器j之间的归一化点积, 即传感器之间的相似度.

选择k个这样的归一化点积组成邻接矩阵, k为滑动窗口大小, 邻接矩阵通过式 (4) 计算获得:

$$A_{ji} = 1\{j \in TopK(\{e_{ki} : k \in C_i\})\} \tag{4}$$

其中, $TopK$表示在归一化点积e_{ji}中选取k个较大值, 当$A_{ji}=1$时表示传感器i与传感器j存在依赖关系.

2.4 时间依赖性学习模块

对S_{trn}按照时间顺序进行滑动窗口划分, 在时刻t往前选取时间戳长度为k的数据, 每一窗口表示为$Q^{(t)} = \left\{S^{(t-k+1)}, S^{(t-1)}, \cdots, S^{(t)}\right\}$, 通过式 (5) 学习时间序列数据的时间特性:

$$F_{\text{Encoder\_out}} = Encoder\left(Q^{(t)}\right) \tag{5}$$

其中, $Q^{(t)}$表示输入数据, $Encoder$表示编码器, $F_{\text{Encoder_out}}$表示经过编码器之后的输出数据.

通过式 (6) 提取重要的时间序列数据特征:

$$H = (\partial)_{i,j} = Norm(F_{\text{Encoder\_out}} M_k^{\mathrm{T}}) \tag{6}$$

$$F_{\text{exattn\_out}} = HM_v \tag{7}$$

其中, M是一个独立于输入的学习参数, 作为整个训练集的记忆单元, 使用两个不同的记忆单元M_k和M_v作为key和$value$, $(\partial)_{i,j}$表示第i个节点和M_k的第j行的相似度, H表示推导出的注意力参数, $F_{\text{exattn_out}}$表示经过外部注意力机制输出数据.

将$F_{\text{exattn_out}}$映射到一组潜在变量Z中, 通过式 (8) 将潜在变量映射回输入空间作为重构结果:

$$M^{(t)} = Decoder(Z) \tag{8}$$

其中, $M^{(t)}$表示经过解码器重构后的数据.

2.5 图自适应注意力预测模块

在时刻t, 模型输入维度变换后的$X^{(t)}$, 其中k表示滑动窗口, 目标是判断在当前时刻值$M^{(t)}$:

$$X_i^{(t)} := [M^{(t-k)}, M^{(t-k+1)}, \cdots, M^{(t-1)}] \tag{9}$$

其中, $X_i^{(t)}$表示传感器i在时刻t的输入数据.

考虑到传感器之间的关系, 采用图注意力机制来融合传感器特征V_i, 通过式 (10) 得到传感器i特征Z_i. 其中, $\alpha_{i,j}$表示传感器i与传感器j之间注意力机制系数, D表示训练得到的权重矩阵, $N(i)$表示关系矩阵A得到节点i的邻接集合:

$$Z_i^{(t)} = ReLU\left(\alpha_{i,i} DX_i^{(t)} + \sum_{j \in N(i)} \alpha_{i,j} DX_j^{(t)}\right) \tag{10}$$

注意系数计算如下:

$$g_i^{(t)} = V_i \oplus DX_i^{(t)} \tag{11}$$

$$\pi(i,j) = LeakyReLU\left(a^{\mathrm{T}}\left(g_i^{(t)} \oplus g_j^{(t)}\right)\right) \tag{12}$$

$$N(i) = \left\{j, A_{ji} > 0\right\} \tag{13}$$

$$\alpha_{i,j} = \frac{\exp(\pi(i,j))}{\sum\limits_{k \in N(i) \cup (i)} \exp(\pi(i,k))} \tag{14}$$

其中, $\oplus$表示串联, $g_i^{(t)}$表示将传感器特征V_i与相应变换后特征$DX_i^{(t)}$连接起来, a表示注意力机制的学习系数向量.

经过以上公式得到N个节点的特征表示, 并通过式 (15) 得到在t时刻的预测值:

$$\hat{M}^{(t)} = Linear([V_1 \circ Z_1^{(t)}, \cdots, V_N \circ Z_N^{(t)}]) \tag{15}$$

其中, ∘表示节点特征与相应的时间序列V_i按元素相乘, $\hat{M}^{(t)}$表示在t时刻的预测值.

由于本文的数据集都是数值型, 所以采用 smooth_l1_loss 作为损失函数:

$$L_{\text{smooth\_l1\_loss}} = \frac{1}{T_{\text{train}} - w} \sum_{t=w+1}^{T_{\text{train}}} \left\| \hat{M}^{(t)} - M^{(t)} \right\|_2^2 \tag{16}$$

2.6 计算图偏差分数模块

通过式 (17) 得到传感器i在时刻t预测值与观测到的值之间的偏差:

$$Err_i(t) = |M_i^{(t)} - \hat{M}_i^{(t)}| \tag{17}$$

其中, $Err_i(t)$表示在时刻t传感器i预测值与观测值存在的偏差.

不同的传感器的偏差可能存在一定的差距, 通过式 (18) 对每个传感器的偏差做归一化处理:

$$a_i(t) = \frac{Err_i(t) - \tilde{u}_i}{\tilde{\omega}_i} \tag{18}$$

其中, $\tilde{u}_i$表示$Err_i(t)$值的中位数, $\tilde{\omega}_i$表示值$Err_i(t)$的四分位范围数值, $a_i(t)$表示时刻t传感器i的标准化值.

因为异常只会影响一小部分传感器, 甚至是单个传感器, 所以为了计算时刻的整体异常情况, 使用函数对传感器进行聚合:

$$y(t) = \max_i a_i(t) \tag{19}$$

如果$y(t)$超过设定的阈值, 就将t时刻的数据标记为异常. 由于图结构模型涉及的参数较多, 为了减少模型的复杂度, 本文阈值采用简单平均值SMA来进行设定:

$$\varepsilon = ave\left(\sum_{t \in T} y(t)\right) \tag{20}$$

其中, ε表示阈值, 当异常分数$y(t) > \varepsilon$, 则视为异常, 否则视为正常数据.

2.7 算法详情

根据图 2 所示, 步骤 1: 传感器数据编码模块对训练数据S_{trn}中的每个传感器用嵌入向量表示, 并送入图结构学习模块学习传感器依赖. 步骤 2: 图结构学习模型首先学习某个传感器除了自身之外与其他传感器之间的关系, 为了避免数据的冗余性, 归一化后, 选择k个这样的归一化点积, 以此组成邻接矩阵. 步骤 3: 对S_{trn}按照一定的批量大小、滑动窗口大小将时间序列数据转换成是三维数据, 送入时间依赖性学习模块学习时间依赖性, 时间依赖性学习模块将$Q^{(t)}$首先送入到以全连接层架构的自编码器中, 强迫神经网络学习在时间维度上最有信息量的特征, 在编码器与解码器之间加入外部注意力机制, 该注意力机制通过对特征图的每个位置都分配一个权重, 这个权重是由外部记忆单元和输入之间的相似性计算得出的. 记忆单元使用了两个线性层M_k、M_v以及归一化层实现的一个独立于输入的参数, 能够作为整个训练数据集的记忆, 具有正则的作用以及提高泛化的能力. 步骤 4: 将两个模块从不同角度学习到的特征送到图注意力预测网络中训练, 在该网络使用 ReduceLROnPlateau 调度器来自动调整学习率, 能够监控验证损失, 并在损失停止下降时降低学习, 这种方法可以自动调整学习率, 以提高模型的性能, 最后得到某个时刻下预测值. 步骤 5: 在图偏差分数计算模块中, 根据观测值和预测值得到偏差分数, 使用Max函数对偏差分数进行聚合得到异常分数, 高于阈值的异常分数判断为异常. 综上所述, 给出 AEEA-GDN 模型构建算法如算法 1 所示.

算法 1. AEEA-GDN 模型构建

1) 样本数据S_{trn}中用V_i表示每个传感器用嵌入向量;
2) 根据特征和特征数N建立候选关系矩阵, 再利用候选关系构造有向边, 构造好的边计算图数据并进行归一化得到e_{ji}, 取前k个最大值组成邻接矩阵A_{ji};
3) S_{trn}按照一定的批量大小、滑动窗口大小转换成是三维数据$Q^{(t)}$, 并送入由外部注意力机制组成的自编码器中学习时间依赖性, 得到$M^{(t)}$;
4) 将图结构特征A_{ji}与时间依赖特征$M^{(t)}$共同送入图注意力预测网络训练;
5) 根据观测值和预测值得到偏差分数, 将偏差分数聚合得到异常分数, 如果异常分数高于阈值, 则判定为异常, 输出二分类结果$a(t)$.

3 实验与结果分析

3.1 实验设置

为了验证本文所提出模型的有效性, 将在时间序列数据集上进行实验. 在文本数据集方面, 选取了 3 个公开可用的数据集, 详见表 1. 这些数据集均来源于真实世界, 涵盖了不同领域的时间序列数据. 下面对这些时间序列数据集进行简要描述.

表 1 异常检测数据集

数据集	维度	训练数据	测试数据	异常率 (%)
SWaT	51	496 800	449 919	11.98
WADI	112	1 048 571	172 801	5.99
MSL	55	58 317	73 729	10.72

安全水处理 (SWaT)[27]: 从水处理试验台 (一个小型网络物理系统) 收集的 11 天多变量时间序列数据.

最近 4 天的数据包含 36 次攻击. 这些攻击的目标和持续时间各不相同.

配水 (WADI)[28]: 从配水管道收集的 16 天多变量时间序列数据. 每个系列包括各种网络流量, 传感器和执行器测量. 16 天中, 正常情况下有 14 天有数据, 攻击场景下有 2 天有数据.

火星科学实验室漫游车 (MSL)[29]: 从火星科学实验室漫游车记录的多变量时间序列数据. 训练试验台和测试试验台是分开的, 测试试验台的异常都被标记.

3.2 模型设置及评价指标

本文的 AEEA-GDN 模型, 由 PyTorch 1.7.0 架构, 使用 CUDA 10.1 版本和 PyTorch 几何库 1.7.0 版本, 在 2 个 NVIDIA RTX 2080Ti 显卡的服务器上训练数据集, 使用 Adam 优化器. 对于 WADI (SWaT、MSL) 数据集, 本文模型使用嵌入向量长度为 128 (64、64)、批量大小为 128 (128, 128)、滑动窗口大小为 30 (15、15) 和 128 (64、64) 神经元的隐藏层. 自编码器由全连接层架构, 隐藏层节点的个数设置为前一层神经元数量的 1/2. 此外, 本文 ReduceLROnPlateau 调度器中的因子参数是 0.1, 耐心参数是 10.

对于性能评估, 由于本文的数据集是二分类的, 所以本文采用 3 个标准的评估指标精确率 (P)、召回率 (R)、$F1$ 分数 ($F1$):

$$\begin{cases} P = \dfrac{TP}{TP+FP} \\ R = \dfrac{TP}{TP+FN} \\ F1 = 2\cdot\dfrac{P\times R}{P+R} \end{cases} \tag{21}$$

其中, TP 是真阳性, FP 是假阳性, FN 是假阴性, ×表示准确率和召回率相乘, P 为检测的准确率, 表示检测到的异常样本的百分比, R 为召回率, 表示正确识别出来的异常样本的百分比, R 越高表示漏报的异常越少, $F1$ 值兼顾准确率和召回率, 是评估模型异常检测性能的主要指标, 只要一个窗口包含的点被检测到异常, 窗口就被标记为异常.

3.3 实验结果与分析

为了评估本研究所提出方法的实际效果, 选择了 3 种典型异常检测算法进行比较, 并以精确率 (P)、召回率 (R)、$F1$ 分数 ($F1$), 作为异常检测性能的评判指标. 在 3 组真实数据集上, 7 个模型的测试结果详见表 2.

表 2 异常检测性能比较 (%)

方法	SWaT			MSL			WADI		
	P	R	$F1$	P	R	$F1$	P	R	$F1$
LSTM-VAE	98.39	77.01	86.40	52.57	95.46	67.80	46.32	32.20	37.99
MSCRED	98.43	77.69	86.64	68.83	88.54	77.45	30.26	40.35	34.58
MAD-GAN	98.72	77.60	86.90	85.17	89.91	87.47	41.44	33.92	37.30
DAGMM	90.60	80.72	85.38	54.12	99.34	70.07	22.28	19.76	20.94
USAD	98.70	74.02	84.60	64.51	86.72	85.31	15.99	51.68	24.93
GDN	92.47	66.24	77.21	78.84	99.19	88.02	26.24	45.94	33.39
本模型	99.63	64.22	**89.12**	82.47	99.06	**90.04**	29.34	61.39	**39.76**

为了研究每个组件的必要性, 采用逐渐排除这些组件的方法来检测模型的性能, 可以直观地看到在缺少相关组件时, 性能是如何下降的. 消融实验对比结果如表 3 所示.

表 3 消融实验对比结果 (%)

方法	SWaT			MSL			WADI		
	P	R	$F1$	P	R	$F1$	P	R	$F1$
AEEA-GDN	99.63	64.22	89.12	82.47	99.06	90.04	29.34	61.39	39.76
–自适应学习	99.19	64.97	78.50	82.22	97.43	89.16	28.72	41.68	33.99
–外部注意力机制	99.71	63.26	77.43	78.84	99.93	88.17	26.24	45.94	33.39
–自编码器	92.47	66.24	77.21	78.84	99.19	88.02	23.95	53.17	33.02

为了直观展示本文 AEEA-GDN 模型的整体性能, 分别在 SWaT、MSL、WADI 数据集上将 P、R、$F1$ 结果对比, 比较结果如图 3–图 5 所示.

对于 AEEA-GDN 的训练过程, 探寻每个组件对模型的影响, 尤其在加入自适应学习组件后, 模型的训练迭代次数能够达到 100 轮次, 得到充分训练. 分别在 3 个数据集上测试, 测试结果如图 6–图 14 所示. 在模型中引入自适应学习之后, 能够改善模型中的分类效果不佳的问题. 图 8、图 11、图 14 中能够看出 acu 逐渐偏向 1, 表示分类效果逐渐改善. 其中, acu 表示分类效果评价指标.

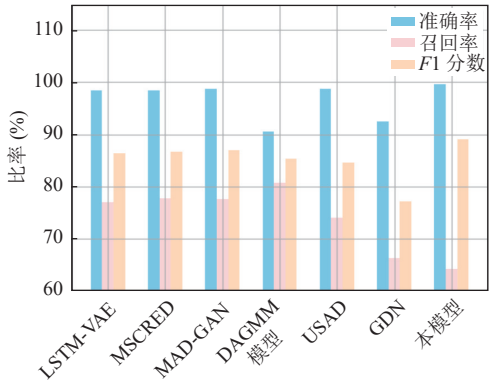

图 3 SWaT 上模型性能比较

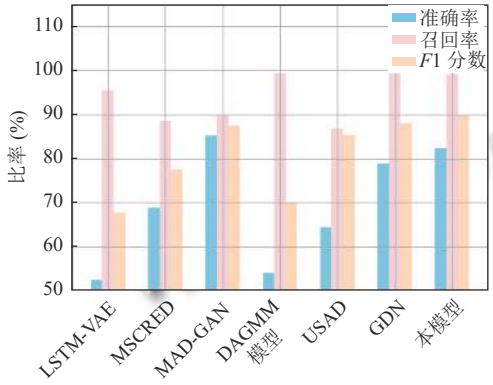

图 4 MSL 上模型性能比较

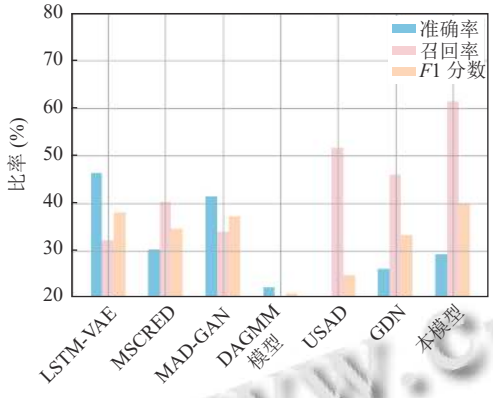

图 5 WADI 上模型性能比较

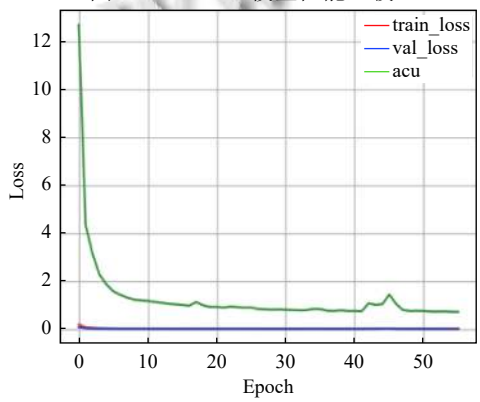

图 6 SWaT 在 GDN 模型变化曲线

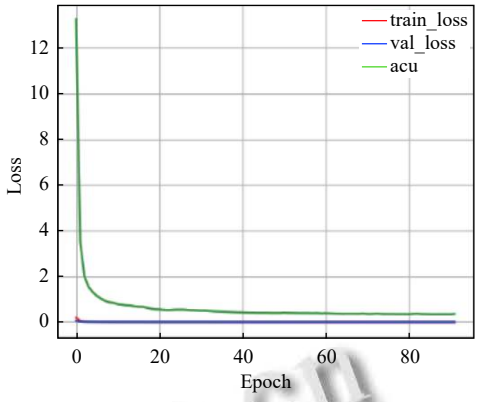

图 7 SWaT 自编码器和外部注意力机制变化曲线

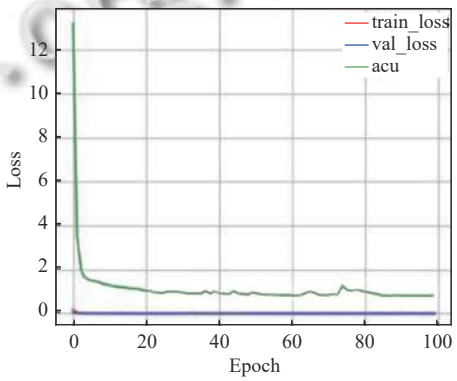

图 8 SWaT 引入自适应学习变化曲线

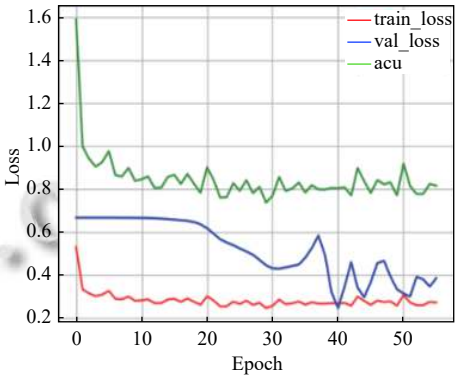

图 9 MSL 在 GDN 模型变化曲线

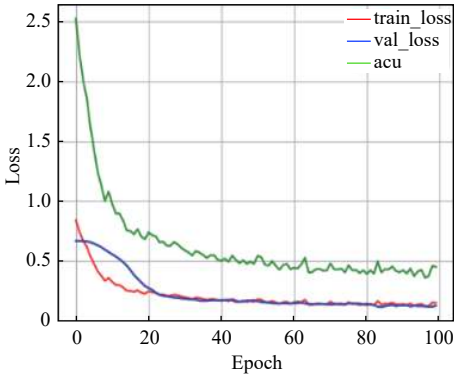

图 10 MSL 自编码器和外部注意力机制变化曲线

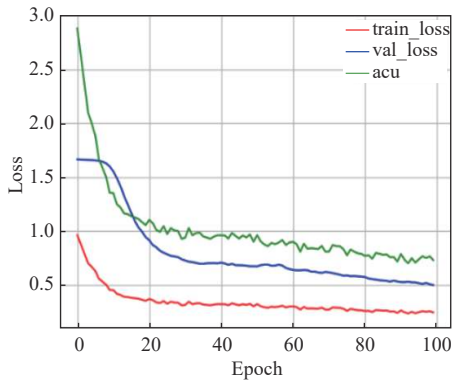

图 11 MSL 自适应学习变化曲线

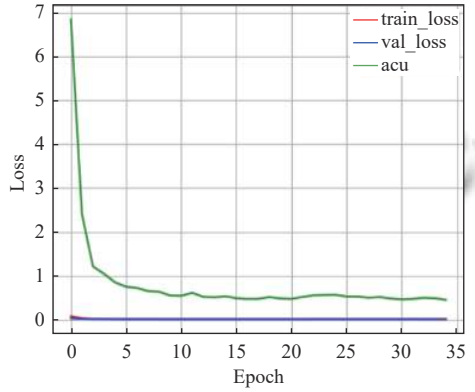

图 12 WADI 在 GDN 模型变化曲线

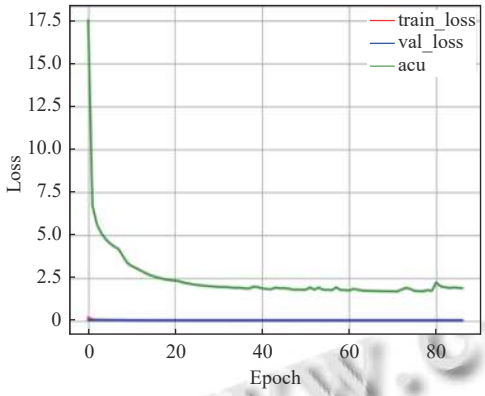

图 13 WADI 自编码器和外部注意力机制

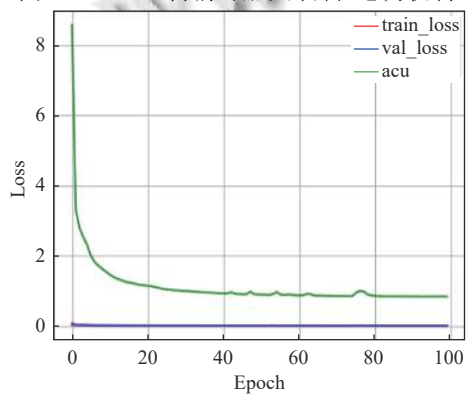

图 14 WADI 数据集引入自适应学习变化曲线

4 结论与展望

针对基于图偏差网络的时间序列异常检测模型, 本文提出将传感器的依赖性和时间依赖性结合送入图预测网络训练, 并且在模型训练时, 加入自适应学习帮助网络更好地适应异常数据的变化, 根据异常数据的变化调整网络的学习率, 在多个公共数据集上的实验结果表明本文模型在异常检测领域的有效性. 在 WADI 数据集上, 本模型训练结果并不很好, 主要是 WADI 数据集涵盖了众多传感器数据, 导致数据维度较高, 这给模型性能带来了挑战. 在未来工作中, 考虑维数超过 100 的情况下, 如何提升本文的异常检测模型的准确率.

参考文献

1 Pang GS, Shen CH, Cao LB, *et al*. Deep learning for anomaly detection: A review. ACM Computing Surveys, 2022, 54(2): 38.

2 Elsken T, Metzen JH, Hutter F. Neural architecture search: A survey. The Journal of Machine Learning Research, 2019, 20(1): 1997–2017.

3 Blázquez-García A, Conde A, Mori U, *et al*. A review on outlier/anomaly detection in time series data. ACM Computing Surveys (CSUR), 2021, 54(3): 1–33.

4 Blázquez-García A, Conde A, Mori U, *et al*. A review on outlier/anomaly detection in time series data. ACM Computing Surveys, 2022, 54(3): 56.

5 Shyu ML, Chen SC, Sarinnapakorn K, *et al*. A novel anomaly detection scheme based on principal component classifier. Proceedings of the 2003 IEEE Foundations and New Directions of Data Mining Workshop. IEEE, 2003. 172–179.

6 Angiulli F, Pizzuti C. Fast outlier detection in high dimensional spaces. Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery. Helsinki: Springer, 2002. 15–27.

7 Schölkopf B, Platt JC, Shawe-Taylor J, *et al*. Estimating the support of a high-dimensional distribution. Neural Computation, 2001, 13(7): 1443–1471. [doi: 10.1162/089976601750264965]

8 Breunig MM, Kriegel HP, Ng RT, *et al*. LOF: Identifying density-based local outliers. ACM SIGMOD Record, 2000, 29(2): 93–104. [doi: 10.1145/335191.335388]

9 Aggarwal CC. An introduction to outlier analysis. Outlier Analysis. 2nd ed., Cham: Springer, 2017. 1–34. [doi: 10.1007/978-3-319-47578-3_1]

10 Zong B, Song Q, Min MR, *et al*. Deep autoencoding

Gaussian mixture model for unsupervised anomaly detection. Proceedings of the 6th International Conference on Learning Representations. Vancouver: OpenReview.net, 2018. 1–19.
11 Audibert J, Michiardi P, Guyard F, *et al*. USAD: Unsupervised anomaly detection on multivariate time series. Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. ACM, 2020. 3395–3404.
12 Li D, Chen DC, Jin BH, *et al*. MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks. Proceedings of the 28th International Conference on Artificial Neural Networks. Munich: Springer, 2019. 703–716.
13 Lample G, Ballesteros M, Subramanian S, *et al*. Neural architectures for named entity recognition. Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. San Diego: Association for Computational Linguistics, 2016. 260–270.
14 Defferrard M, Bresson X, Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering. Proceedings of the 30th International Conference on Neural Information Processing Systems. Barcelona: Curran Associates Inc., 2016. 3844–3852.
15 Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. Proceedings of the 5th International Conference on Learning Representations. Toulon: OpenReview.net, 2016.
16 Veličković P, Cucurull G, Casanova A, *et al*. Graph attention networks. Proceedings of the 6th International Conference on Learning Representations. Vancouver: OpenReview.net, 2018.
17 Deng AL, Hooi B. Graph neural network-based anomaly detection in multivariate time series. Proceedings of the 35th AAAI Conference on Artificial Intelligence. AAAI, 2021. 4027–4035. [doi: 10.1609/aaa.v35i5.16523]
18 Gori M, Monfardini G, Scarselli F. A new model for learning in graph domains. Proceedings of the 2005 IEEE International Joint Conference on Neural Networks. IEEE, 2005. 729–734.
19 Yu B, Yin HT, Zhu ZX. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. Proceedings of the 27th International Joint Conference on Artificial Intelligence. Stockholm: IJCAI.org, 2017. 3634–3640.
20 Lim N, Hooi B, Ng SK, *et al*. STP-UDGAT: Spatial-temporal-preference user dimensional graph attention network for next POI recommendation. Proceedings of the 29th ACM International Conference on Information & Knowledge Management. ACM, 2020. 845–854. [doi: 10.1145/3340531.3411876]
21 Wang YW, Wang W, Ca YJ, *et al*. Detecting implementation bugs in graph convolutional network based node classifiers. Proceedings of the 31st IEEE International Symposium on Software Reliability Engineering (ISSRE). Coimbra: IEEE, 2020. 313–324.
22 Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. Nature, 1986, 323(6088): 533–536.
23 Vaswani A, Shazeer N, Parmar N, *et al*. Attention is all you need. Proceedings of the 31st International Conference on Neural Information Processing Systems. Red Hook: Curran Associates Inc., 2017. 6000–6010.
24 Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. Proceedings of the 3rd International Conference on Learning Representations. San Diego: ICLR, 2015.
25 Wang XL, Girshick R, Gupta A, *et al*. Non-local neural networks. Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 7794–7803.
26 Guo MH, Liu ZN, Mu TJ, *et al*. Beyond self-attention: External attention using two linear layers for visual tasks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 45(5): 5436–5447.
27 Mathur AP, Tippenhauer NO. SWaT: A water treatment testbed for research and training on ICS security. Proceedings of the 2016 International Workshop on Cyber-physical Systems for Smart Water Networks. Vienna: IEEE, 2016. 31–36.
28 Ahmed CM, Palleti VR, Mathur AP. WADI: A water distribution testbed for research in the design of secure cyber physical systems. Proceedings of the 3rd International Workshop on Cyber-physical Systems for Smart Water Networks. Pennsylvania: ACM, 2017. 25–28.
29 Hundman K, Constantinou V, Laporte C, *et al*. Detecting spacecraft anomalies using LSTMs and nonparametric dynamic thresholding. Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. London: ACM, 2018. 387–395. [doi: 10.1145/3219819.3219845]

(校对责编: 牛欣悦)