

文章编号:2095-6134(2023)01-0109-10

基于 GAN 序列的卫星数据接收系统故障定位*

王正晟^{1,2†}, 李亚林¹, 张洪群¹

(1 中国科学院空天信息创新研究院, 北京 100094; 2 中国科学院大学电子电气与通信工程学院, 北京 100049)
(2020 年 10 月 30 日收稿; 2021 年 2 月 5 日收修改稿)

Wang Z S, Li Y L, Zhang H Q. Satellite data receiving system fault location based on GAN sequence[J]. Journal of University of Chinese Academy of Sciences, 2023, 40(1): 109-118. DOI:10.7523/j.ucas.2021.0009.

摘要 现有的卫星数据接收系统故障定位方法受困于专家知识的难以获取。提出一种基于生成对抗网络(GAN)序列的故障定位方法 GANseq。利用故障传播特性,将接收系统分解为 M 个信号处理单元(SPU),形成一个按照信号处理顺序排列的 SPU 序列;将故障定位问题转化为有序排列的 M 个异常检测子问题,其中第 m 个子问题为对前 m 个 SPU 的联合状态的异常检测;然后针对每个子问题采用基于 GAN 的状态参数检测模型,进而形成 GAN 序列;综合分析该序列在所有子问题上的检测结果以定位故障 SPU。利用实际接收系统进行的实验证明所提出的 GANseq 不仅从数据驱动的角度完成了故障定位,而且在总体上降低了虚警率,并提高了定位精度。

关键词 卫星数据接收系统;故障定位;生成对抗网络;数据驱动
中图分类号:TN957 **文献标志码**:A **DOI**:10.7523/j.ucas.2021.0009

Satellite data receiving system fault location based on GAN sequence

WANG Zhengsheng^{1,2}, LI Yalin¹, ZHANG Hongqun¹

(1 Aerospace Information Research Institute, Chinese Academy of Sciences, Beijing 100094, China;
2 School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract Existing satellite data receiving system fault location methods suffer from the difficulty in obtaining expert knowledge. A generative adversarial network (GAN) sequence method called GANseq is proposed in this paper. According to the feature of fault propagation, the receiving system was first divided into M signal processing unit (SPU) to form a SPU sequence ranked by signal processing order (SPO). The fault location issue then was decomposed into ranked M anomaly detection sub-problems, where the m th sub-problem was the detection of the top m SPUs' joint state. State parameters GANomaly-based detector was employed in each sub-problem, forming a GAN sequence. The detection results of this sequence on all sub-problems were analyzed to locate fault SPU. The experiment result from practical receiving system shows that GANseq can not only achieve fault location from data-driven perspective, but also reduce fault alarm rate and enhance accuracy level of fault location.

Keywords satellite data receiving system; fault location; GAN; data-driven

* 中国科学院战略先导科技专项 A 类(XDA15000000)资助
† 通信作者, E-mail: wangzs@aircas.ac.cn

卫星数据接收系统利用天线对经过无线传输的卫星高频微波信号进行接收后,通过各个信号处理环节将其处理为二进制数据。接收系统的故障定位就是在故障发生时确定发生故障的信号处理环节。由于接收系统组成复杂且各信号处理环节相互耦合^[1],难以对其建立精确的故障定位模型。目前已有的故障定位方法大致分为 3 类:基于专家系统(expert system, ES)的方法^[2]、基于故障树分析(fault tree analysis, FTA)的方法^[3-9]、基于人工神经网络(artificial neural networks, ANNs)的方法^[10]。

ES 方法利用专家知识根据故障现象制定故障定位所需要的条件和结果,形成故障定位规则存入数据库。在实际使用中其推理机通过规则匹配确定故障源。

FTA 方法将基于二叉树构建的故障树模型引入推理机的设计,成为目前接收系统故障定位的主流方法。由根节点到叶节点的层次和逻辑关系对应于故障树的推理过程。因为二叉树数据结构可以方便地实现节点的匹配与遍历、插入与删除,所以缓解了规则的冲突和冗余。文献[5]利用案例推理(case based reasoning)加快了 FTA 的推理速度。文献[6]用包含自动化测试的 FTA 检测方法降低了故障树叶节点的虚警率,减少了故障排查时间。文献[7]将对接收流程的检测纳入了故障定位的范畴以延展 FTA 方法的检测范围。文献[8]为克服 FTA 方法的计算复杂度问题,将双向联想记忆(bidirectional associative memory)网络用于定位故障源。因为通常无法在故障树中找到偶发故障的处置方法,文献[9]用各种故障模式之间的关系构建了接收系统的模糊 Petri 网(fuzzy Petri net),并结合维护经验和资源的流动推理出故障源。

上述两个方法均在对接接收系统建模的基础上进行故障定位。与此相对,利用 ANNs 对复杂系统强非线性特性的拟合能力,可从数据驱动的角度尝试故障定位。郎国伟等^[10]讨论了反向传播(back propagation, BP)网络在接收系统的故障定位中的应用,将故障定位问题转化为多目标分类问题。首先分别在每个信号处理环节出现故障的情况下实测各监测点的信号电平,总结出代表正常状态和各个环节发生故障的异常状态的信号电平特征向量,并用其训练 BP 网络分类器。训练后的分类器可以根据异常电平向量分类出其对应

的故障环节,从而实现故障定位。

前述方法虽然都在一定程度上完成了定位工作,但是它们有两个方面的缺陷:1) ES、FTA 还有它们的拓展方法都是专家知识方法的变体。它们将接收系统的各部分之间的耦合信息融入故障定位规则或者故障树的结构以实现信号处理环节之间的解耦。但当涉及到不同设备之间或复杂设备的内部环节之间的耦合关系时,相应的专家知识难以获取,这使得该类方法的虚警率较高、定位精度低并且难以适应系统的更新升级和设备更换;2) ANNs 方法虽然从信号电平特征向量本身出发完成了故障定位,但是为了使向量能够代表整个系统的状态从而使训练出的分类器达到较高的分类准确度,监测点的选取工作仍需大量的专家知识,所以该方法并没有真正意义上实现数据驱动。除此之外,由于接收系统在通常情况下处于正常运行状态,训练分类器所需的异常样本极难获得。当不同种类样本的数量不平衡时,分类器会通过牺牲少数样本来保证多数样本的分类准确度,这意味着训练后的分类器在检测阶段会偏向于做出系统无故障的判断。

在执行卫星数据接收任务期间,所有信号处理环节的状态参数都会被记录并整合为状态参数向量存入运行日志。日志所蕴含的大量数据为使用深度学习的方法解决故障定位问题提供了基础。基于此,本论文提出基于生成对抗网络(generative adversarial network, GAN)序列的故障定位方法 GANseq。该方法能够解决数据接收系统样本数量不平衡的问题,进而实现数据驱动的故障定位。具体地,首先:本文引入信号处理单元(signal processing unit, SPU)的概念,基于对接接收系统故障传播特性方面的研究,将信号处理环节划分到各个 SPU 中,从而将接收系统分解为一个按照信号处理顺序排列的 SPU 序列,并将 SPU 作为故障源进行定位。其次:针对每个 SPU 以及可能对其状态造成影响的其他 SPU,构建基于 GAN 的检测模型用于检测它们的联合状态,进而形成一个 GAN 序列。最后:在训练阶段,仅用正常状态参数训练每个 GAN 模型;在故障定位阶段,综合所有模型对相应 SPU 的联合状态的检测结果以确定故障 SPU,从而完成故障定位。

本方法的主要贡献包括两点:1) 实现数据驱动的故障定位:GAN 检测模型在训练和检测中均只需信号处理环节的状态参数,而无需建立复杂

的故障诊断模型。此外,由于训练后的模型相比于专家知识能够更加全面、细致地反映接收系统在相应区域的特性,所以在完全避免专家知识获取环节的同时也降低了虚警率;2)提升故障定位精度:GANseq 首次将 SPU 分解方法引入接收系统故障定位中,将很多原本不能被独立分析的信号处理环节映射于不同的 SPU,进而从整个系统中分离出故障环节,从总体上提升了故障定位的精度。

1 GANseq 方法概述

GANseq 的流程如图 1 所示。首先引入 SPU 的概念,将 SPU 定义为信号处理环节的集合。随

后在对接收系统故障传播特性研究的基础上提出 SPU 分解方法,将接收系统的所有信号处理环节映射至 SPU 中,从而将接收系统分解为 M 个不可分割且硬件独立 SPU,形成一个按照信号处理顺序排列的 SPU 序列。本文用 SPU_m 表示序列中的第 m 个 SPU。GANseq 以 SPU 作为故障源进行定位,利用故障状态在 SPU 序列中的传播特点,将接收系统的故障定位问题分解为 M 个有序排列的异常检测子问题,其中第 $m(m=1,2,\cdots,M)$ 个子问题对应于前 m 个 SPU 的联合状态的检测。在故障定位中,若前 $m-1$ 个 SPU 的联合状态的检测结果正常,而前 m 个 SPU 异常,则可判定 SPU_m 为故障源。

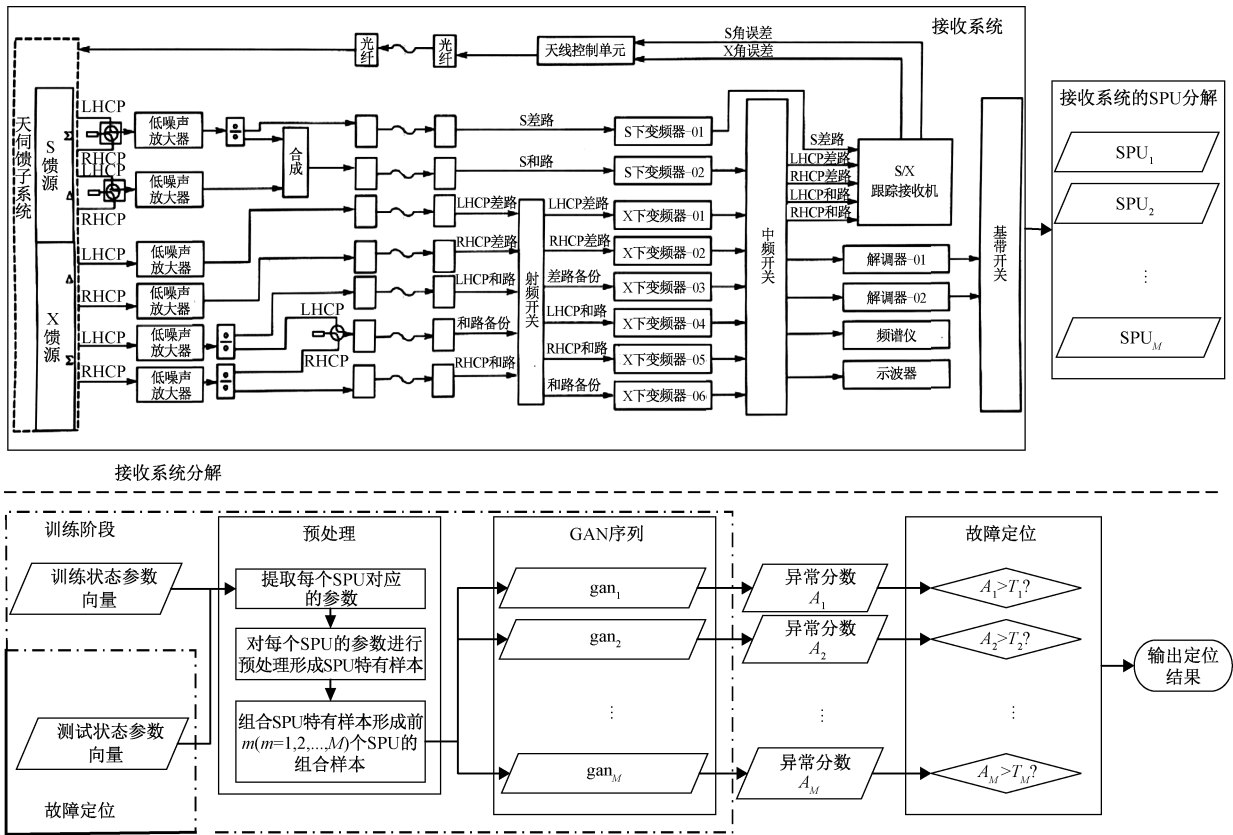


图 1 GANseq 技术路线
Fig. 1 GANseq Technical Route

运行日志中的大量状态参数为应用基于深度学习的数据驱动方法解决每个异常检测子问题提供了基础。由于接收系统的故障率较低,异常样本获取困难,所以 GANseq 通过引入 GAN 来应对由此引发的不平衡数据集问题。鉴于图像检测模型 GANomaly^[11] 的优良性能,GANseq 在每个子问题 m 中应用了基于 GANomaly 构建的检测器 gan_m , 形成一个 GAN 序列。具体地,用前 m 个 SPU 的正常状

态参数训练 gan_m , 训练完成后的 gan_m 在检测中输出待测状态参数向量中前 m 个 SPU 对应的参数的异常分数 A_m , 随后通过 A_m 是否超过阈值 T_m 判定前 m 个 SPU 的联合状态是否异常。

2 理论

2.1 故障传播特性

由于不同设备之间、单个设备内部信号处理

环节之间均存在耦合性,一个环节发生故障可能导致其他环节的状态出现异常。以往的定位方法通常依赖于维护经验和专业知识解耦信号处理环节,对于无法解耦的环节,只能将其整合为一个故障源进行定位。在这种情况下,专家知识掣肘了定位性能。

通过对故障传播特性的研究可知,多个信号处理环节之间的耦合关系源于信号的流动:在形成有效数据之前,接收系统需要依次调用不同的信号处理环节对同一卫星射频信号进行处理,被调用的环节按照信号处理顺序的方向形成了一个信号处理环节序列。任一中间环节的故障导致其输出信号发生的异常会影响后续环节的状态。耦合关系的这种诱因意味着故障在信号处理环节之间的传播方向与接收信号的流向相同,即使能够检测出某个环节的状态参数异常也不能确定其本身为故障源。

但另一方面,信号处理环节的故障状态并不会影响先于其对信号进行处理的其他环节。利用这一点,可以对一个调用了 C 个信号处理环节的接收任务使用如下故障定位策略。

策略 1:首先按照信号处理顺序将接收任务分解为信号处理环节 1 至 C ,然后依次设计状态参数检测模型检测前 $c(c=1,2,\dots,C)$ 个环节的联合状态,如果前 $c-1$ 个环节的联合状态正常而前 c 个环节的联合状态异常,就可以确定第 c 个环节为故障源。

2.2 接收系统的 SPU 分解

由于不同卫星的工作模式可能不同,不同卫星的接收任务对应的信号处理环节序列不尽相同。如果使用策略 1,则必须针对不同卫星的接收任务单独设计联合状态检测模型,而如今卫星的工作体制种类繁多,这种方法没有可行性。

为解决这一问题,引入 SPU 的概念,并提出接收系统的 SPU 分解方法,利用接收系统分解产生的 SPU 序列将所有卫星接收任务的信号处理环节序列统一起来。以 SPU 为故障源,针对每个 SPU 和能够对其状态产生影响的其他 SPU 的联合状态设计唯一的检测模型以实现故障定位。

2.2.1 SPU 分解

SPU 被定义为信号处理环节的集合,它的故障状态被定义为其所包含的任意一个信号处理环节出现故障的状态。通过将信号处理环节按照一定的规则映射到 SPU 中从而将接收系统分解为

一个按照信号处理顺序排列的 SPU 序列。

映射规则包含以下 2 条:1)建立从信号处理环节到 SPU 的“多对一”映射,使得不同的 SPU 不包含相同的信号处理环节,从而保证 SPU 之间的硬件独立性,使得 SPU 可以作为独立故障源进行定位;2)将在不同卫星接收任务的信号处理顺序中两两之间相对位置不确定的信号处理环节映射至同一 SPU,并将 2 个 SPU 之间的相对位置定义为任意两个分别被映射于此二者的信号处理环节在所有卫星接收任务信号处理顺序中的相对位置。由于所有相对位置不确定的环节都在同一 SPU 内,所以属于不同 SPU 的 2 个信号处理环节在信号处理顺序中具有唯一的相对位置关系,从而将接收系统分解为了一个按照信号处理顺序排列的 SPU 序列。

原则 1)和原则 2)共同保证了 SPU 序列的唯一性,而且还使得序列中的每个 SPU 都不可被再次分割,从而将故障定位的粒度最小化,保证了定位的精度。

2.2.2 针对 SPU 序列的故障定位

由于 SPU 序列的方向与信号处理顺序的方向相同,所以类比于故障在信号处理环节间的传播特性,故障状态在 SPU 间的传播方向也与接收系统分解产生的 SPU 序列的方向相同——故障 SPU 在 SPU 序列中只会影响其后续 SPU 的状态,而对其之前的 SPU 不会产生影响。假设接收系统分解产生的 SPU 序列含有 M 个 SPU,基于故障状态在 SPU 序列中的这种传播特性,GANseq 将策略 1 推广至 SPU 序列得到故障定位策略 2,将其用于定位故障 SPU。

策略 2:以 SPU 为故障源。将接收系统的故障定位问题分解为 M 个有序排列的异常检测子问题,其中第 $m(m=1,2,\dots,M)$ 个子问题对应于前 m 个 SPU 的联合状态的检测,并依此设计相应的检测模型。在对任何卫星的接收任务进行故障定位时,依次检测 SPU 序列中前 m 个 SPU 的联合状态。如果发现前 $m-1$ 个 SPU 的联合状态正常,而前 m 个 SPU 的联合状态异常,就可以判定故障源为 SPU_m 。

3 方法

如图 1 所示,GANseq 包含 3 个步骤:1)接收系统的 SPU 分解;2)GAN 序列的构造和训练;3)故障定位。在将接收系统分解为含有 M 个 SPU

的 SPU 序列后,根据策略 2,需要在每个子异常检测问题 m 中构建以前 m 个 SPU 的状态参数为输入的异常检测模型从而能够根据前 m 个 SPU 的状态参数判断其联合状态异常与否。因为运行日志中含有异常状态参数的向量较少,所以在每个子问题中构建结构相似的 GAN 检测模型。

3.1 GAN 序列的构造和训练

3.1.1 基于 GANomaly 的检测模型

异常检测就是在一类数据(正常样本)远多于另一类数据(异常样本)的不平衡数据集中将异常样本从整体中区分出来。虽然异常检测可以被直接当作监督学习问题处理,但是由于日志数据中的正常状态参数向量占比极大,所以代表各信号处理环节故障的异常向量的规模无法达到能够被有效拟合的程度。由此,GANseq 采用单类型、半监督的学习范式只对正常向量进行拟合。Goodfellow 等^[12]提出的 GAN 已经成为该范式的首要方法^[13-16]。这其中,Akca 等^[11]在双向生成对抗网络(bidirectional GAN, BiGAN)^[16]的基础上推出的 GANomaly 框架在图像检测方面的表现优异。GANomaly 由编码器 G_E 和 E 、解码器 G_D 、判断网络 D 组成:编码器 G_E 和解码器 G_D 组成了一个类似于自动编码器的结构,作为整个模型的生成网络 G ;编码器 E 将生成网络的输出再次压缩至隐含空间(latent space) \mathcal{Z} ;判别网络 D 在训练中鉴别真实图像和 G_D 输出的伪造图像;在它的基础上构建每个异常检测子问题 m 的检测模型 gan_m (结构如图 2 所示)。并对原始结构进行 2 处修改以使模型适配于运行日志中的数据:1)将原

网络中的卷积层(convolution layer)和转置卷积层(transpose convolution layer)替换为全连接层(fully connected layer)以使之可以接收一维样本作为输入;2)原解码器 G_D 采用了与原编码器 G_E 不同的结构,而我们在 gan_m 中使解码器 $G_{D,m}$ 与编码器 $G_{E,m}$ 的结构镜像对称,以形成一个典型的自动编码器,更好地适应一维样本的编码与重构^[17]。

由于 gan_m 需对前 m 个 SPU 的联合状态进行检测,所以它的输入样本 \mathbf{x}_m 是由状态参数向量中前 m 个 SPU 所包含的信号处理环节的状态参数经过预处理得到的一维向量。在训练阶段,解码器 $G_{D,m}$ 通过学习从隐含空间 \mathcal{Z} 构造正常样本以达到拟合正常状态参数的概率分布的目的,同时编码器 $G_{E,m}$ 学习成为 $G_{D,m}^{-1}$ 并和编码器 E_m 一同显式地学习 \mathcal{Z} 的概率分布。将输入样本 \mathbf{x}_m 的异常分数 $A(\mathbf{x}_m)$ 定义为 \mathbf{x}_m 经过 $G_{E,m}$ 和 E_m 两次编码分别形成的隐含向量 $G_{E,m}(\mathbf{x}_m)$ 和重构隐含向量 $E_m(G_m(\mathbf{x}_m))$ 之间的欧式距离:

$$A(\mathbf{x}_m) = \| G_{E,m}(\mathbf{x}_m) - E_m(G_m(\mathbf{x}_m)) \|_2. \quad (1)$$

因为生成网络在训练中只拟合了正常样本的内部结构,它的参数不适合产生异常样本,所以当代表前 m 个 SPU 中存在故障的异常样本 \mathbf{x}_m 被编码器 $G_{E,m}$ 映射为隐含向量 \mathbf{z}_m 时,解码器 $G_{D,m}$ 无法重构这个样本。生成网络 G_m 此时的输出 $G_m(\mathbf{x}_m)$ 被剔除了异常特征,这致使编码器 E_m 编码得到的重构隐含向量与原始隐含向量之间存在较大差异、产生较大的异常分数 $A(\mathbf{x}_m)$,从而使得模型可以将异常样本 \mathbf{x}_m 甄别出来。

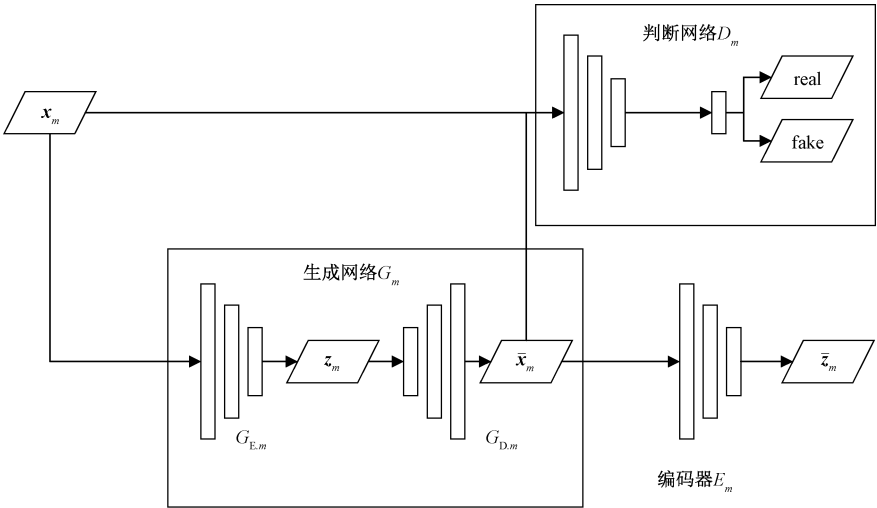


图 2 gan_m

Fig. 2 gan_m

3.1.2 预处理

GAN 模型通常希望输入值为取值范围在 0 到 1 之间的数值类型,并且在 GANseq 中,每个检测模型的输入样本对应于 SPU 序列的不同子序列,所以在将状态参数向量送至 GAN 序列进行故障定位前需要对其进行预处理(如图 1 中的“预处理”方框所示)。

假设需要进行故障定位的接收任务的状态参数向量用 \mathbf{P} 表示。预处理工作首先将 \mathbf{P} 中属于每个 SPU 的参数提取出来,形成 M 个 SPU 特有参数向量 \mathbf{P}_m 。

然后,在特有参数向量 \mathbf{P}_m 上针对不同类型的参数采用不同的处理方式,以形成 SPU 特有样本 \mathbf{f}_m 。状态参数向量中的数据分为二元、标称、数值 3 种类型,接下来分类型叙述预处理方法:1) 二元类型:取值为 0 或 1,已处于 0~1 区间内,不需要进行处理;2) 标称类型:可能的取值为数量有限的离散整数,需要将每个标称数据独热编码^[18]为一个长度为取值种类数的比特序列;3) 数值类型:包含普通型和累积型 2 种。对于普通型直接正则化至 0~1 区间即可。累积数值类型参数的值为该参数从信号处理开始直到参数采集时刻的累积量,所以需要将其与上一采集时刻的值做差以得到采样时刻的增量,然后再正则化至 0~1 区间;

此处需要注意的是,在正则化的过程中必须根据信号处理环节的使能开关二元参数略过没有实际参与信号处理的闲置环节的无效状态参数值,这是因为闲置环节的参数随机取值,而随机值会偏离有效值的取值范围,从而造成有效值被正则化至一个很小的区间内影响模型的拟合。除此之外,任由 \mathbf{P}_m 中的无效参数被输入检测模型会引起模型在训练阶段的发散、故障定位阶段的误检,因此在正则化工作结束后将特有样本 \mathbf{f}_m 中没有参与正则化的随机值置零以消除它们的状态信息,如此一来,训练后的模型能够在检测中自动屏蔽闲置信号处理环节所携带的噪声。

最后,将 M 个 SPU 特有样本 \mathbf{f}_m 组合为 M 个检测器的输入样本。 gan_m 的输入样本 \mathbf{x}_m 为 SPU₁ 至 SPU _{m} 的特有样本 \mathbf{f}_1 至 \mathbf{f}_m 的组合。如下所示

$$\mathbf{x}_m = [\mathbf{f}_1^T, \mathbf{f}_2^T, \dots, \mathbf{f}_m^T]^T. \quad (2)$$

3.1.3 训练流程

gan_m 的训练使编码器 $G_{E,m}$ 和 E_m 能够将样本

中的正常成分映射进隐含空间 \mathcal{Z} ,而 $G_{D,m}$ 可以根据 \mathcal{Z} 中的正常成分重构原始样本,从而达到剔除异常成分的目的。所以训练 $G_{E,m}$ 、 $G_{D,m}$ 、 E_m 的损失函数 L_g 包含对抗损失、语境损失、异常分数 3 部分^[11]:

对抗损失 L_{adv}

$$L_{\text{adv}} = \|D_{m,f}(\mathbf{x}_m) - D_{m,f}(G_m(\mathbf{x}_m))\|_2. \quad (3)$$

其中 $D_{m,f}$ 是判断网络 D_m 倒数第 2 层的等效输出函数。

语境损失 L_{con}

$$L_{\text{con}} = \|\mathbf{x}_m - G_m(\mathbf{x}_m)\|_1. \quad (4)$$

综上所述,损失函数 L_g 为

$$L_g = \omega_{\text{adv}} L_{\text{adv}} + \omega_{\text{con}} L_{\text{con}} + \alpha A(\mathbf{x}_m). \quad (5)$$

其中 ω_{adv} 、 ω_{con} 和 α 是权重参数。

用判断误差 L_{fraud} 训练判断网络 D_m

$$\begin{aligned} L_{\text{fraud}} &= \sigma(D_m(\mathbf{x}_m), 1), \\ L_{\text{fraud}} &= \sigma(D_m(G_m(\mathbf{x}_m)), 0). \end{aligned} \quad (6)$$

其中 σ 为交叉熵。

3.2 故障定位步骤

在进行定位前,先取一定数量的正常状态参数向量组成分数阈值计算集。对于检测前 m 个 SPU 状态的子问题,首先预处理阈值计算集中的向量产生相应的样本 \mathbf{x}_m 。然后用训练好的模型 gan_m 依次计算样本 \mathbf{x}_m 的异常分数,并取其中的最大值作为在定位中用于判断前 m 个 SPU 的联合状态异常与否的分数阈值 T_m 。

在定位中,GAN 序列中的每个检测模型 gan_m 依次为待进行故障定位的接收任务的状态参数向量中相应 SPU 的参数打分,得到异常分数序列 $[A_1, A_2, \dots, A_M]$ 。然后遍历序列,判断分数是否大于相应阈值,若前 $m-1$ 个 SPU 对应的异常分数 $A_{m-1} \leq T_{m-1}$ 而前 m 个 SPU 对应的异常分数 $A_m > T_m$ 则可判定故障源为 SPU _{m} ;倘若没有任何异常分数超出阈值,则可认为接收任务正常执行;

4 实验验证

为验证 GANseq 的有效性,用其对某实际运行的接收站的数据接收系统进行故障定位实验。同时,该站目前使用的基于 FTA 和自动化测试的故障诊断 ES(以下简称“原方法”)被用作对比对象。

实验采用的数据取自该站的运行日志。将用于实验的状态参数向量集划分为 20 000 个训练

向量、1 000 个阈值计算向量、6 987 个测试向量,测试向量由 5 987 个代表每个特定信号处理环节故障的异常状态参数向量和 1 000 个正常状态参数向量

组成。按照信号处理顺序将接收系统分解为包含 22 个 SPU($M=22$)^[19] 的 SPU 序列(表 1)。SPU 序号增大的方向为信号处理顺序的方向。

表 1 接收系统分解
Table 1 Receiving system decomposition

设备	信号处理环节	SPU 序号	设备	信号处理环节	SPU 序号
低噪声放大器	15 V 电源 A	1	中频开关	电源 A	13
	15 V 电源 B			电源 B	
	电流		VCC 电压		
	放大器	2	2.5 V 电源	14	
光发射机	电源	3	解调器	1.2 V 电源	
	发射放大器	4		载波恢复	15
	发射功率	5		混频	16
光接收机	电源	6		低通滤波	17
	接收功率	7		时钟恢复	18
	接收放大器	8		抽样判决	19
射频开关	电源 A	9		维特比译码	20
	电源 B			帧同步	21
下变频器	5 V 电源	10		RS 译码	
	12 V 电源			LDPC 译码	22
	本振锁定	11		解扰	
	变频	12			

因为设备的电源、电流、电压信号处理环节为支撑其余环节的辅助环节,其正常工作是其他环节处于正常状态的必要条件,又因为其本身不处理信号从而其运行状态不受其他环节影响,所以其所属的 SPU 位于相应设备的最前端。由于所有辅助环节均为控制整个设备的全局性环节,它们在信号处理顺序中没有确定的相对位置关系,所以将一个设备的所有辅助环节划归于同一 SPU。

由于不同的卫星对信号进行加扰和编码的顺序可能不同,所以将解调器内部的解扰环节、LDPC 译码环节和 RS 译码环节划归于同一 SPU (SPU₂₂)。

4.1 模型的训练

用 tensorflow2.0 框架实现 GAN 序列中的每个模型 gan_m。gan_m 的结构如表 2 所示。inputs 是模型输入样本 \mathbf{x}_m 的长度。编码器 G_{E,m} 将 \mathbf{x}_m 压缩至维度为 hides 的隐含空间 \mathcal{Z} 。在训练中用 Adam 和 SGD 优化器分别优化生成网络和判断网络^[20]。两者均采用指数学习率衰减方案(exponential scheduling),其中初始学习速率为 0.01,衰减速率为 0.1。以 256 个样本为一批次(batch-size = 256),训练 gan_m 1 000 个迭代次数(epochs = 1 000)。此外,对于 L_g 中的 3 个权重参数,首先采用 GridSearchCV^[18] 进行网格搜索(grid

search)和交叉验证(cross validation),然后根据最小生成网络损失函数 L_g 决定其最优取值。

以检测 SPU₁ 至 SPU₂₁ 联合状态的模型 gan₂₁ 为例。分别绘出代表 SPU₂₁ 故障的异常测试样本 $\bar{\mathbf{x}}_{21}$ 的隐含向量 $\mathbf{z}=G_{E,21}(\mathbf{x}_{21})$ 和重构隐含向量 $\mathbf{z}_1=E_{21}(G_{21}(\mathbf{x}_{21}))$,并取正常测试样本 \mathbf{x}_{21} 与之对照,如图 3 所示。不难发现,由于 \mathbf{x}_{21} 在编码和重构的过程中被去掉了异常成分, \mathbf{x}_{21} 的隐含向量的重构偏离其本体,而 \mathbf{x}_{21} 的隐含向量和其重构高度吻合。这说明被定义为隐含向量和隐含向量重构之间欧式距离的异常分数能有效地反映异常参数

表 2 gan_m 结构
Table 2 Structure of gan_m

子网络	层	神经元数
编码器 G _{E,m}	Input	inputs
	Dense(LeakyRelu, Batchnorm)	hides
	Dense(LeakyRelu)	hides
解码器 G _{D,m}	Input	hides
	Dense(LeakyRelu, Batchnorm)	hides
	Dense(tanh)	inputs
编码器 E _m	Input	inputs
	Dense(LeakyRelu, Batchnorm)	hides
	Dense(LeakyRelu)	hides
判别网络	Input	inputs
	Dense(LeakyRelu, Batchnorm)	hides
	Dense(LeakyRelu, Batchnorm)	hides
	Dense(sigmoid)	1

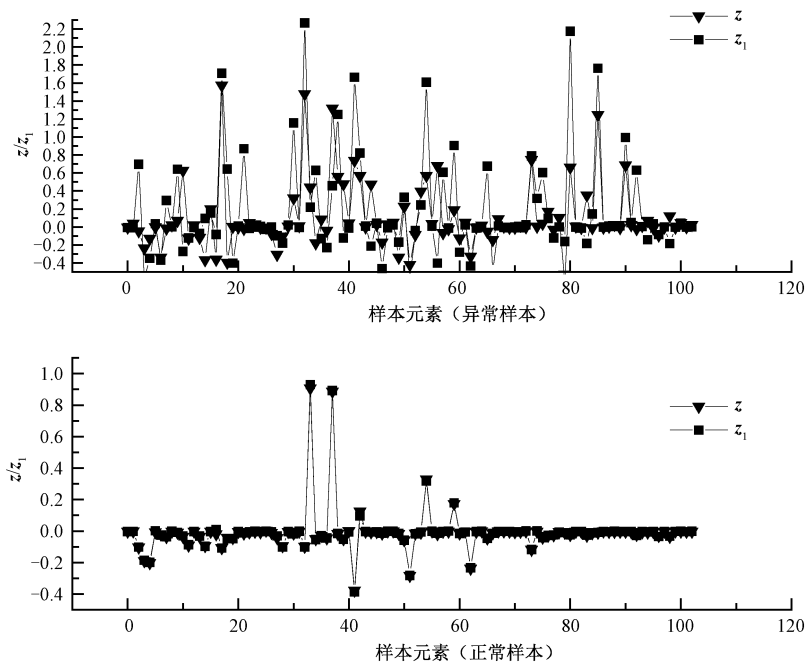


图 3 隐含向量和重构隐含向量

Fig. 3 Latent vector and reconstructed latent vector

相对于正常参数概率分布的偏离程度,对异常情况有较强的指示性。

4.2 评价指标

利用虚警率(fault alarm rate, FAR)和召回率(recall, R)比较两种方法在每个故障源上的定位性能。定义虚警率 FAR 和召回率 R 如下:

$$FAR = \frac{F_p}{T_p + F_p}, \tag{7}$$

$$R = \frac{T_p}{T_p + F_N}. \tag{8}$$

对于某故障源而言: T_p 是该故障源的异常测试向量集中被正确定位的向量个数; F_p 是正常测试向量集中被错误定位为该故障源的向量个数(虚警个数); F_N 是该故障源的异常测试向量集中没有被正确定位的向量个数; 召回率 R 越大, 相应方法对于该故障源发生故障的敏感性越强。需要注意的是, 对于 GANseq 而言故障源为 SPU; 对于原方法而言故障源为信号处理环节; 根据 SPU 故障状态的定义, SPU 故障源的异常测试向量集为该 SPU 包含的所有信号处理环节故障源的异常测试向量集的并集。

4.3 实验结果

测得原方法和 GANseq 的虚警率 FAR(表 3 的 3、4 列)和召回率 R(表 3 的 6、7 列)。作为对 GANseq 的补充, 当定位出的 SPU 对应多个辅助

信号处理环节时, 对相应的辅助环节进行进一步的点检(“GANseq+点检”), 即检查电源状态指示位是否异常、电压值或电流值是否超过正常限度。“GANseq+点检”的 FAR 和 R 分别如表 3 的第 5、8 列所示。

由于解调器结构复杂, 故障树难以拟合其内部信号处理环节的耦合关系, 所以原方法没有对解调器内部的环节进行定位, 只是监控了电路板上的温控系统和电平值, 并且在实际运行中, 原方法在只有一个监控点报警的情况下就会做出解调器异常的警告, 所以本实验计算其在解调器的所有非辅助信号处理环节整体上的虚警率和召回率。

与原方法相比, GANseq 将除解调器和辅助信号处理环节之外的环节上的定位虚警率 FAR 平均降低 49.8%。由于 GANseq 在将接收系统分解为 SPU 序列的过程中通过保证 SPU 的不可分割性最小化了故障定位粒度, 所以绝大多数 SPU 只包含一个信号处理环节, 并且在加入点检之后, “GANseq+点检”在辅助环节上的虚警率也与原方法基本持平, 所以 GANseq 在实现数据驱动故障定位的同时从整体上降低了虚警率。

GANseq 将解调器内部大多数原本作为整体定位的信号处理环节按照映射规则一对一地映射至 SPU 中, 并且以 70% 以上的召回率和 14% 以下

表 3 实验结果
Table 3 Experiment result

故障源		FAR/%			R/%		
		原方法	GANseq	GANseq+点检	原方法	GANseq	GANseq+点检
低噪声放大器	15 V 电源 A	9.1		0.9	91.8		98.5
	15 V 电源 B	7.3	0.5	0.8	83.1	99.2	99.3
	电流	1.9		0.9	89.9		98.9
	放大器	68.9	4.7	4.7	69.1	96.7	96.7
光发射机	电源	5.6	7.8	7.8	88.5	93.4	93.4
	发射放大器	79.1	10.5	10.5	58.9	93.1	93.1
	发射功率	72.1	3.3	3.3	65.7	95.5	95.5
光接收机	电源	8.1	14.7	14.7	94.1	87.5	87.5
	接收功率	75.3	10.6	10.6	56.7	90.2	90.2
	接收放大器	70.7	9.0	9.0	61.2	91.2	91.2
射频开关	电源 A	7.3		9.6	85.4		87.3
	电源 B	2.0	8.6	7.7	82.9	85.4	88.9
下变频器	5 V 电源	4.8		13.8	82.4		82.2
	12 V 电源	2.2	13.7	12.5	79.8	79.2	78.1
	本振锁定	62.4	0.6	0.6	73.4	82.8	82.8
	变频	77.9	10.7	10.7	77.5	81.3	81.3
中频开关	电源 A	0.9		5.1	77.4		77.4
	电源 B	1.9	4.5	3.9	92.6	78.1	76.3
解调器	VCC 电压	5.5		8.2	73.6		79.3
	2.5 V 电源	2.9	7.2	7.6	80.3	78.1	76.4
	1.2 V 电源	10.8		6.9	89.5		79.6
	载波恢复		10.0	10.0		70.9	70.9
	混频		13.7	13.7		70.5	70.5
	低通滤波	79.1	9.9	9.9	78.2	73.1	73.1
	时钟恢复		13.9	13.9		73.9	73.9
	抽样判决		9.1	9.1		74.2	74.2
	维特比译码		12.9	12.9		73.7	73.7
	帧同步		0.9	0.9		70.2	70.2
	RS 译码						
	LDPC 译码		2.2	2.2		74.0	74.0
	解扰						

的虚警率实现了它们的故障定位,提高了故障定位的精度。

原方法的召回率 R 不受信号流动的影响,而 GANseq 的 R 值沿着信号处理顺序的方向呈现下降趋势。GANseq 出现这一趋势的原因在于其在定位的过程中只要检测出故障 SPU 就立即输出定位结果,所以如果在真正的故障 SPU 之前产生了误检,那么定位工作会提前结束从而导致定位失败,而误检的累积使得在信号处理顺序中的位置越靠后的 SPU 受到误检的影响越明显、具有更小的召回率。

与“GANseq+点检”相比,原方法在辅助信号处理环节上获得较低虚警率的同时保证了较高的召回率。这归因于辅助环节结构简单,原方法在故障树推理中对其所进行的点检就可以精确地检测出异常状态。但是由于“GANseq+点检”中的

点检阶段在 GANseq 的基础上进行,所以其在辅助环节上的召回率受制于 GANseq 在包含该环节的 SPU 上的召回率,从而导致“GANseq+点检”在位置靠后的某些辅助环节上的 R 值低于原方法。

5 结论

本文提出基于 GAN 序列的卫星数据接收系统故障定位方法 GANseq。该方法利用接收系统故障传播特性将整个接收系统的故障定位问题按照信号处理顺序分解为多个基于 GAN 的异常检测问题,全面地利用地面站已有的运行状态数据,实现数据驱动的实时故障定位,摆脱了一直困扰传统定位方法的专家知识难以获取问题,并且从整体上降低了故障源的虚警率、提高了故障定位的精度。但是,GANseq 目前面临着两个问题: 1) 由于 GANseq 在故障定位中一旦检测到故障源

即停止检测,所以其在每次定位工作中只能定位出一个故障源,与此同时,定位工作的提前结束也会造成后续故障源的漏检;2) GANseq 不能直接分离少数在信号处理顺序中相对位置不恒定的信号处理环节,而只能将它们组合为 SPU 作为整体定位。在未来的工作中将会着力解决这些问题。

参考文献

[1] 冯钟葵,张洪群,王万玉,等. 遥感卫星数据获取与处理关键技术概述[J]. 遥感信息, 2008, 23(4): 91-97, 75. DOI:10.3969/j.issn.1000-3177.2008.04.019.

[2] 朱杰,林维夏,赵现纲. 气象卫星数据接收远程故障智能诊断系统开发[C]//中国气象学会. 第35届中国气象学会年会论文集: S21 卫星气象与生态遥感. 合肥, 2018: 267-268.

[3] 靳芳,张洪群,葛小青. 卫星接收故障诊断专家系统的研究[J]. 微计算机信息, 2009, 25(7): 107-108, 99.

[4] 韦春龙. 卫星接收系统故障诊断专家系统的研究[J]. 信息网络安全, 2012(2): 81-83. DOI:10.3969/j.issn.1671-1122.2012.02.024.

[5] 陈丹,胡涛,王建涛,等. 基于 CBR 和 FTA 的地面站接收系统故障诊断方法[J]. 太赫兹科学与电子信息学报, 2019, 17(2): 227-233.

[6] 申鸿烨. 基于监测点与自动化测试的接收系统故障诊断[J]. 控制工程, 2016, 23(3): 417-422. DOI:10.14107/j.cnki.kzgc.150486.

[7] 韩威,路宽,年昭华,等. 遥感卫星接收系统故障诊断系统的研究[C]//第四届高分辨率对地观测学术年会论文集. 武汉, 2017: 1153-1165.

[8] 陈丹,胡涛,王建涛,等. 基于故障树与 BAM 网络的卫星地面站接收系统故障诊断方法[J]. 信息工程大学学报, 2018, 19(3): 301-305. DOI:10.3969/j.issn.1671-0673.2018.03.009.

[9] 蒋超,张政治. 基于扩展故障树和模糊 Petri 网的导航地面监测站故障诊断方法[C]//第十一届中国卫星导航年会论文集: S08 测试评估技术. 成都, 2020: 78-81.

[10] 郎国伟,周东方,胡涛,等. 基于遗传算法和神经网络的

故障诊断研究[J]. 信息工程大学学报, 2017, 18(2): 140-142. DOI:10.3969/j.issn.1671-0673.2017.02.003.

[11] Akcay S, Atapour-Abarghouei A, Breckon T P. GANomaly: semi-supervised anomaly detection via adversarial training[C]// Computer-Vision-ACCV2018. Lecture Notes in Computer Science, 2019, 11363. DOI:10.1007/978-3-030-20893-6-39. <https://arxiv.org/abs/1805.06725>.

[12] Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative adversarial networks [J]. Communications of the ACM, 2020, 63(11): 139-144. DOI:10.1145/3422622.

[13] 吴春志,冯辅周,吴守军,等. 一种有效的不平衡样本生成方法及其在行星变速箱故障诊断中的应用[J]. 兵工学报, 2019, 40(7): 1349-1357. DOI:10.3969/j.issn.1000-1093.2019.07.003.

[14] Schlegl T, Seeböck P, Waldstein S M, et al. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery [C] // Information Processing in Medical Imaging. IPMI 2017. Lecture Notes in Computer Science, 10265. DOI:10.1007/978-3-319-59050-9-12.

[15] Donahue J, Krähenbühl P, Darrell T. Adversarial feature learning[EB/OL]. (2017-04-03) [2020-09-10]. <https://www.arxiv.org/abs/1605.09782v7>. arXiv:1605.09782.

[16] Zenati H, Foo C S, Lecouat B, et al. Efficient GAN-based anomaly detection[EB/OL]. (2019-05-01) [2020-09-10]. <https://arxiv.org/abs/1802.06222>. arXiv:1802.06222.

[17] Patel A A. Hands-on unsupervised learning using python: How to build applied machine learning solutions from unlabeled data [M]. Sebastopol, USA: O'Reilly, 2019: 222-232.

[18] Géron A. Hands-on machine learning with scikit-learn, keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems[M]. Sebastopol, USA: O'Reilly, 2019.

[19] 周炯槃,庞沁华,续大我,等. 通信原理[M]. 4 版. 北京: 北京邮电大学出版社, 2015.

[20] Chintala S. How to train a GAN? Tips and tricks to make GANs work [EB/OL]. (2020-03-05) [2020-05-30]. GitHub. <https://github.com/soumith/ganhacks>.