基于深度学习的拟态裁决方法研究

杨晓晗[1]，程国振[1,2]，刘文彦[1,2]，张帅[1]，郝兵[3]

（1. 信息工程大学信息技术研究所，河南 郑州 450002； 2. 网络空间安全教育部重点实验室，河南 郑州 450000；
3. 嵩山实验室，河南 郑州 450046）

摘 要：针对软硬件差异化容易导致拟态裁决结果不一致所造成的假阳现象被误认为网络攻击的问题，提出了一种基于深度学习的拟态裁决方法。通过构建无监督的自编码−解码深度学习模型，挖掘不同执行体输出多样化正常响应数据的深度语义特征，分析归纳其统计规律，并通过设计基于离线学习−在线裁决联动的训练机制和基于反馈优化机制来解决假阳现象，从而准确检测网络攻击，提高目标系统的安全弹性。鉴于软硬件差异导致正常响应数据间的统计规律已被深度学习模型理解掌握，因此不同执行体间拟态裁决结果将保持一致，即目标系统处于安全状态。一旦目标系统受到网络攻击，执行体的响应数据将偏离深度学习模型的统计规律，致使拟态裁决结果不一致，即目标系统存在潜在安全威胁。实验结果表明，所提方法的检测性能显著优于主流的拟态裁决方法，且平均预测准确度提升了 14.89%，有利于将该方法集成到真实应用的拟态化改造来增强系统的防护能力。

关键词：拟态防御；主动防御；拟态裁决；深度学习；离线训练−在线裁决

中图分类号：TP393.08

文献标志码：A

DOI: 10.11959/j.issn.1000−436x.2024047

Research on mimic decision method based on deep learning

YANG Xiaohan[1], CHENG Guozhen[1,2], LIU Wenyan[1,2], ZHANG Shuai[1], HAO Bing[3]

1. Institute of Information Technology, Information Engineering University, Zhengzhou 450002, China
2. Key Laboratory of Cyberspace Security, Ministry of Education, Zhengzhou 450000, China
3. Songshan Laboratory, Zhengzhou 450046, China

Abstract: Due to software and hardware differentiation, the problem of false positives mistakenly identified as network attack behavior caused by inconsistent mimic decision results frequently occurs. Therefore, a mimic decision method based on deep learning was proposed. By constructing an unsupervised autoencoder-decoder deep learning model, the deep semantic features of diverse normal response data were explored from different executions and its statistical rules were analyzed and summarized. Additionally, the offline learning-online decision-making mechanism and the feedback optimization mechanism were designed to solve false positive problem, thereby accurately detecting network attacks and improving target system security resilience. Since statistical rules of normal response data was understood and mastered by deep learning model, the mimic decision results among different executions could remain consistent, indicating that the target system was in a secure state. However, once the target system was subjected to a network attacks, the response data outputted by the different executions was deviated from statistical distribution of deep learning model. Therefore, inconsistent mimic decision results were presented, indicating that the affected execution was under attack and the target system was exposed to potential security threats. The experiments show that the performance of the proposed method is significantly superior to the popular mimic decision methods, and the average prediction accuracy is improved by 14.89%, which is conducive to integrating the method into the mimic transformation of real application to enhance the system’s defensive capability.

Keywords: mimic defense, active defense, mimic decision, deep learning, offline learning-online decision-making

收稿日期：2023−09−27；**修回日期：**2023−12−20

基金项目：河南省重大科技专项基金资助项目（No.221100211200）

Foundation Item: The Major Science and Technology Project of Henan Province (No.221100211200)

0 引言

随着信息技术的快速发展，越来越多的行业应用都在向数字化、智能化、自动化的方向转变来达到提高效率、降低成本及改善产品和服务质量等目标。因此，这些应用对网络的依赖程度日益加剧，互联网已经成为连接人与人、人与物以及物与物的重要纽带。然而，人们在享受科技飞速发展带来便利的同时，网络空间面临的安全问题也日益严峻，时时刻刻威胁着数字世界的基本秩序和规则[1-2]。特别是，网络攻击、数据泄露等事件频频发生，充分说明了网络空间易攻难守，面临严峻的安全挑战。无论是硬件还是软件，都不可避免地受到设计或者逻辑缺陷导致的未知漏洞或后门等问题，而这些漏洞或后门等安全问题无法从根本上杜绝。因此，入侵者能够利用这些漏洞或者后门获取权限并对软硬件系统进行非法访问甚至破坏。

然而，现有的防御方法大多是基于先验的网络攻击知识或者有经验的安全技术人员的被动式防御方法，比如常用的防火墙、入侵检测、入侵防护等安全手段，其缺点是不能防御未知攻击，且防御架构较为单一，很容易被攻击者掌握内在规律，失去防御的效果[3-6]。因此，为了有效解决未知漏洞后门攻击问题，近年来，网络空间拟态防御（CMD, cyber mimic defense）技术被广泛研究[7-14]。其基本思想是通过构建一个基于动态异构冗余（DHR, dynamic heterogeneous redundancy）的核心架构，将动态、随机和多样的异构执行体引入目标系统，通过功能等价体的动态切换构建不确定性、异构性、非持续性的拟态环境，并通过检测不同执行体之间的一致性来判别目标系统是否遭受网络攻击。

拟态防御的DHR架构如图1所示，主要由输入代理模块、异构执行体集、异构构件集、输出裁决模型和负反馈控制模块组成。输入代理模块将用户的请求作为输入，并将其分发到不同的异构执行体中，且不同的异构执行体的功能等价。异构执行体集通过动态选择方法从异构构件集中构建。针对不同执行体的输出响应，通过构建输出裁决模型，判别输出响应的一致性，进而检测出当前执行体是否受到网络攻击。若输出裁决模型对不同执行体输出响应数据的结果具有一致性，则表明目标系统处于安全状态；反之，则判定相应执行体存在安全威胁，需要启动负反馈控制模块对存在安全威胁的执行体进行清洗轮换，从而维持目标系统的安全性。

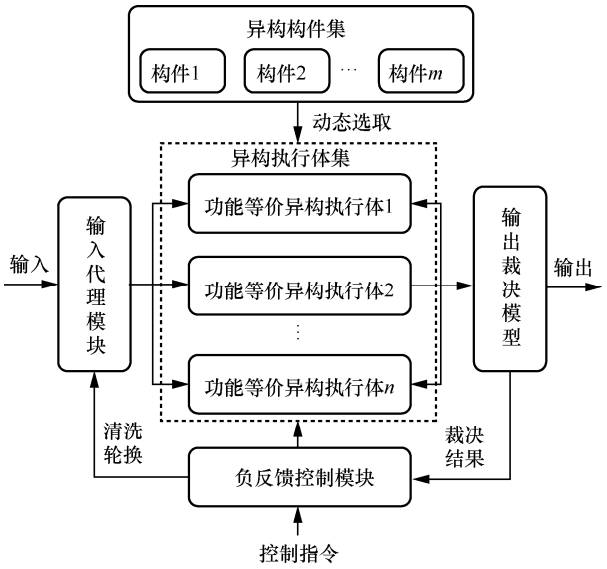

图1 拟态防御的DHR架构

由此可见，输出裁决模型在拟态防御中占有至关重要的地位，能够识别目标系统是否存在安全威胁。但是，目前拟态裁决模块的设计主要聚焦于对多个不同执行体输出响应数据进行逐字节比对输出裁决结果。若不同执行体间字节比对结果完全一致，则认为该系统未受到网络攻击威胁，即全体一致性裁决方法[15]。若多数以上执行体间字节比对结果一致，则认为该系统也未受到网络攻击，即多数裁决方法[16-19]。虽然这些方法对目标系统起到了一定的防御作用，但是无形中加剧了数据间的敏感度，很容易将拟态防御系统中软硬件资源差异导致的不同执行体输出正常响应数据的不一致误判为网络攻击行为，从而造成假阳现象，影响防御效果。

针对此问题，本文提出了一种新颖的拟态裁决方法，主要利用深度学习思想来解决假阳问题。转变主流拟态裁决方法对正常响应数据的敏感度，通过设计简化的自编码−解码深度学习模型，挖掘不同执行体正常响应数据间的深度语义特征，统计归纳正常响应数据间的分布规律，使其扩大正常数据的检测域，并通过与其统计分布的偏离度来准确检测网络攻击的异常行为。本文的主要贡献如下所示。

1) 提出了深度语义特征学习的拟态裁决方法，将数据层面的浅层特征转换到深度学习模型的深度

语义特征，学习不同正常数据之间的相关性关系，扩大正常数据的检测范围，降低主流裁决方法对浅层数据的高敏感度，从而提高攻击检测的准确度。

2) 提出了基于离线学习−在线裁决联动的深度学习模型。离线学习阶段，构建自编码−解码的深度学习模型，学习不同正常响应数据之间的统计规律，并将其作为下一阶段检测正常或异常数据的基准。在线裁决阶段，通过设计与已学习的正常响应数据的统计分布的偏离度，来检测待测响应数据是否出现异常。

3) 提出了一种增强深度学习模型的检测准确度的反馈优化机制，一旦在线裁决阶段输出的结果出现错误，就立即启动该机制，将错误数据反馈上报给深度学习模型，并通过反馈优化校正模型的训练参数，从而进一步增强深度学习模型的检测准确度。

1　基于深度学习的拟态裁决方法设计

1.1　架构设计

基于深度学习的拟态裁决架构如图 2 所示。此架构由 4 个模块组成，包括预处理模块、离线训练模块、在线裁决模块和反馈优化模块。首先，预处理模块对不同执行体响应数据进行预先处理，使其满足深度学习模型的输入要求。其次，考虑到系统未上线时执行体响应数据都是未受网络攻击影响的正常样本，离线训练模块提出的自编码−解码深度学习模型以无监督的训练方式学习不同执行体响应数据之间的统计分布规律，降低主流拟态裁决方法对数据本身的灵敏度。再次，在线裁决模块在系统上线后，将待测执行体响应数据送入已训练好的深度学习模块，推演出待测数据是否符合统计规律，若符合则裁决结果为正常，表明执行体未受攻击影响，反之则为异常，需要通过拟态防御的反馈控制机制对异常数据对应的执行体进行清洗轮换，确保目标系统的安全性。最后，启动反馈优化模块，将错误的裁决结果上报反馈给离线训练模块的深度学习模型来不断优化校准正常数据的统计分布，自适应地提高裁决准确度。

1.2　预处理

由于业务数据类型不同、包体大小也不尽相同等，因此预处理旨在统一不同执行体响应数据的大小，使其满足深度学习模型的输入要求，如图 3 所示。首先，获取大量的基于应用数据作为输入的不同执行体的响应数据。其次，统计不同执行体响应

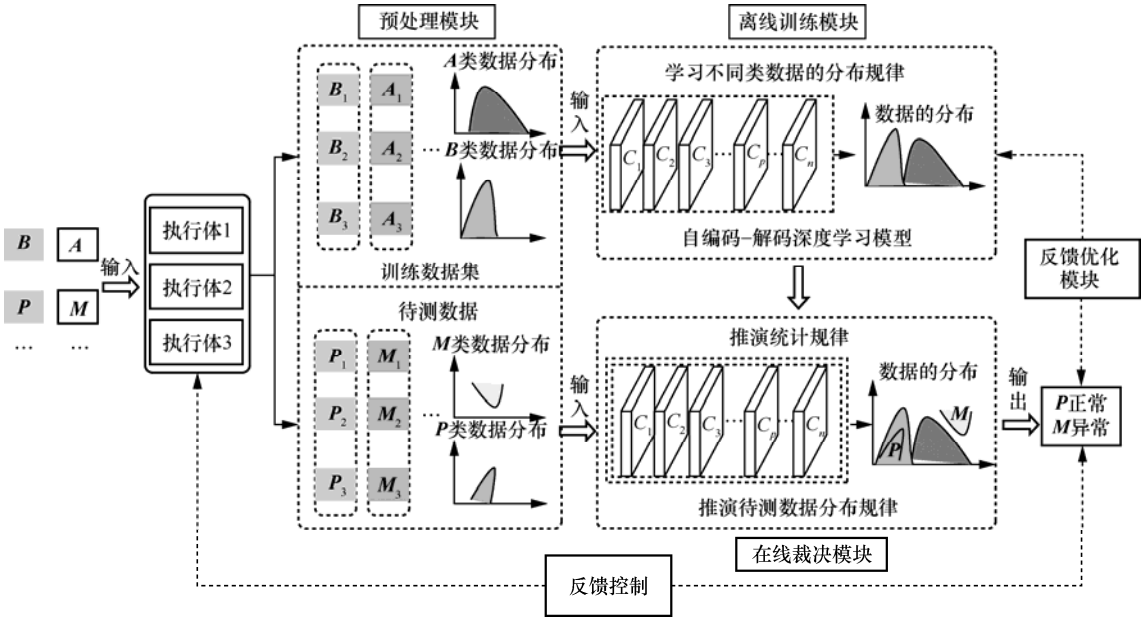

图 2　基于深度学习的拟态裁决架构

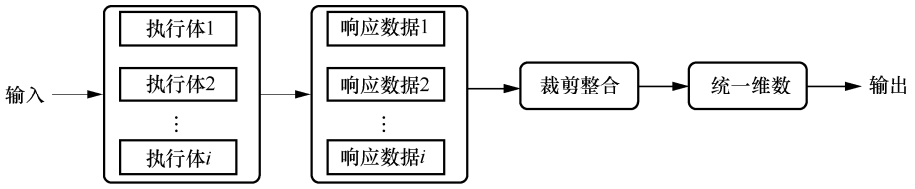

图 3　预处理流程

数据的尺度，选择最大尺度的响应数据作为基准，对不满足此尺度的其他响应数据进行裁剪整合，即对不满足最大尺度要求的响应数据进行补 0，使数据统一到相同的维数作为深度学习模型的输入。

1.3 深度学习模型的网络结构

根据应用拟态化改造的场景特性，本文采用基于无监督的自编码−解码深度学习模型。该模型由两部分构成：自编码模块和解码模块，如图 4 所示。自编码模块由若干全连接（FC, fully connected）层[20]构成，旨在将输入的不同执行体响应数据进行编码，压缩数据的维度去除冗余数据，在潜在空间构建深度语义特征；解码模块也由若干 FC 层构成，旨在将编码后的瓶颈特征进行逐层重塑，达到输入数据和输出数据一致的目标。以五层网络为例，自编码−解码深度学习模型的参数构成如表 1 所示。通过自编码−解码的深度学习模型，能够挖掘不同执行体响应数据的潜在深度语义特征，并自适应地学习不同执行体响应数据分布的内在规律，掌握数据间的统计规律。其优势在于能够将拟态改造过程中资源异构化导致的正常响应数据的不一致性转化为二维空间的统计分布，扩展了正常样本的检测域，进而解决一维空间正常响应数据间高敏感度问题。

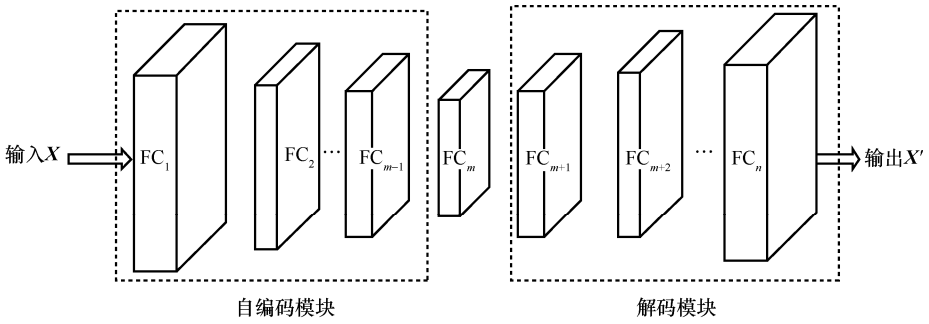

图 4 自编码−解码的深度学习模型

表 1 自编码−解码深度学习模型的参数构成

网络层	输出尺寸	设置
FC_1	1×256	激活函数 ReLU
FC_2	1×25	激活函数 ReLU
FC_3	1×2	激活函数 ReLU
FC_4	1×25	激活函数 ReLU
FC_5	1×256	—

1.4 损失函数

如图 4 所示，自编码−解码的深度学习模型旨在保证输入样本 $\boldsymbol{X}$ 与重塑后的输出样本 $\boldsymbol{X}'$ 之间近似相等，因此，本文采用均方误差（MSE, mean square error）[21]作为深度学习模型的损失函数来最小化输入和输出之间的差异性。损失函数 L 为

$$L(\boldsymbol{X},\boldsymbol{X}')=\frac{1}{M}\sum_{i=1}^{M}(\boldsymbol{X}-\boldsymbol{X}')^2 \tag{1}$$

其中，$\boldsymbol{X}$ 为输入样本，$\boldsymbol{X}'$ 为输出样本，M 为输入样本的个数。

1.5 训练策略

1.5.1 基于离线学习−在线裁决联动的训练机制

根据应用场景拟态化改造需求，提出了基于离线学习−在线裁决联动的训练机制，如图 5 所示，主要分为 2 个阶段：离线学习和在线裁决。首先，在拟态化改造过程中，目标系统处于离线学习阶段，即所有的执行体响应数据都为正常样本，因此利用离线学习的训练方法，通过基于无监督的训练方法训练自编码−解码深度学习模型，使其能够学习到不同执行体输出正常样本数据的统计分布，并根据分布规律设定满足正常样本域的检测阈值。待系统上线后，进入在线裁决阶段，对不同执行体响应输出的待测样本进行测试，即将其送入已训练好的自编码−解码深度学习模型，获取每一个执行体响应数据的 MSE 值，并将不同执行体响应数据的 MSE 值与设定的正常样本阈值进行对比，若待测执行体响应数据的 MSE 值小于或等于设定的阈值，则说明该待测样本符合正常样本域的统计规律，目标系统未受到安全威胁；反之，若待测执行体响应数据的 MSE 值大于设定的阈值，则说明该待测样本偏离了正常样本域的统计规律，目标系统存在安全威胁，需要及时将该待测样本对应的执行体进行上报，并通过拟态防御的反馈控制模块，对此时的异常执行体进行清洗和轮换，从而持续保持目标系统的安全性。

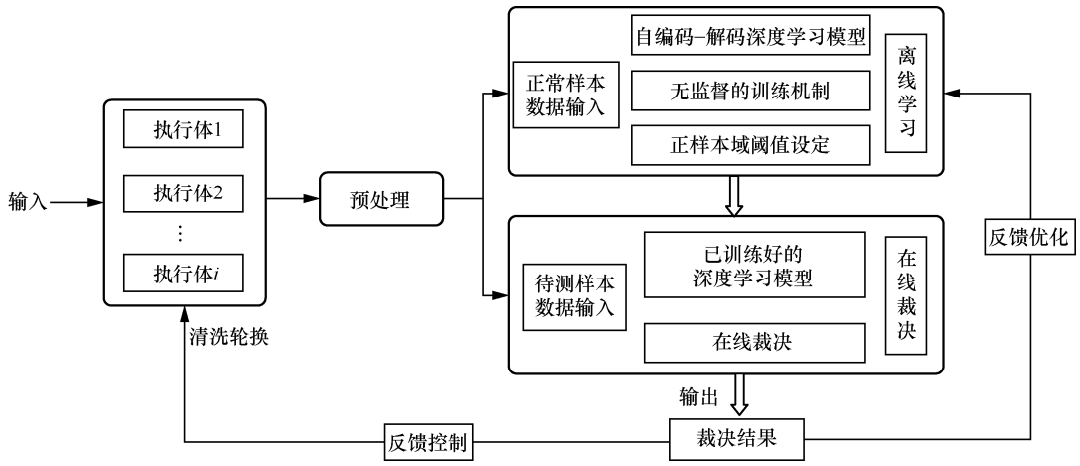

图 5　离线学习–在线裁决联动的训练机制

1.5.2　基于反馈优化机制

基于反馈优化机制设计旨在增强深度学习模型的检测准确度，当在线裁决模块出现错误裁决信息时，能够及时将错误信息反馈给离线学习模型，并通过优化自编码–解码深度学习模型来校正错误的裁决结果，从而提高模型的判别准确度，达到自适应地提高深度学习模型裁决准确度的目的。

2　实验与分析

2.1　实验设置

2.1.1　异构执行体的组成

为了构建一组功能等价、结构各异的多个执行体，本文搭建了异构执行体的开发环境，如表 2 所示。执行体 1 采用 X86+ubuntu20.04 的系统配置，执行体 2 采用 X86+centos7.9 的系统配置，执行体 3 采用 ARM+centos7.9 的系统配置，且不同执行体满足功能等同但结构各异的拟态防御 DHR 架构条件。

表 2　功能等价结构各异的执行体的硬件和软件

执行体	硬件	软件
执行体 1	CPU 架构：X86	ubuntu20.04
执行体 2	CPU 架构：X86	centos7.9
执行体 3	CPU 架构：ARM(鲲鹏 920)	centos7.9

2.1.2　样本集的制作

基于上述 3 个执行体的开发环境，本文随机抓取 6 个真实应用的样本数据，如图 6 所示，并以此为基准模拟仿真了 9 980 个样本来构建训练和测试样本集，其中训练集包含 9 800 个正常样本，测试集包含 180 个样本（90 个正常样本和 90 个异常样本）。由于真实数据中包含 7 个影响因素{code, msg, current, createtime, size, id, total}，训练集通过随机选取 3 个影响因素获取 9 800 个正常样本，测试集中正常样本是通过随机选取剩余 4 个影响因素中的 2 个来获取的，且训练集正常样本和测试集正常样本不重叠；异常样本则通过增加异常字符串和随机修改除正常样本以外的因素来获取。

{"code":200,"msg":"pu63n1YIs5","data":
[{"current":100,"size":200,"id":300,"createtime":"2023-03-13T23:03:50.
670965633-04:00"},
{"current":200,"size":400,"id":500,"createtime":"2023-03-13T23:03:50.
670965891-04:00"}],"total":400}

(a) 样例1

{"code":200,"msg":"wPGyseMZ5T","data":
[{"current":200,"size":200,"id":300,"createtime":"2023-03-13T23:05:20.
376470741-04:00"},
{"current":200,"size":400,"id":500,"createtime":"2023-03-13T23:05:20.
376471015-04:00"}],"total":400}

(b) 样例2

{"code":200,"msg":"xMS22Juhnb","data":
[{"current":2000000,"size":200,"id":300,"createtime":"2023-03-13T23:06:
13.15889453-04:00"},
{"current":3000,"size":400,"id":500,"createtime":"2023-03-13T23:06:13.
158894804-04:00"}],"total":400}

(c) 样例3

{"code":200,"msg":"pR4LFzZvPHU8Ig","data":
[{"current":2000000,"size":400,"id":300,"createtime":"2023-03-13T23:07:
31.007296899-04:00"},
{"current":3000,"size":200,"id":500,"createtime":"2023-03-13T23:07:31.
007297156-04:00"}],"total":400}

(d) 样例4

{"code":200,"abt":"123ew2","msg":"6p7yHC2hY2zh1K","data":
[{"current":2000000,"size":400,"id":300,"createtime":"2023-03-13T23:10:
06.752265912-04:00"},
{"current":3000,"size":200,"id":500,"createtime":"2023-03-13T23:10:06.
752266144-04:00"}],"total":400}

(e) 样例5

{"code":200,"abt":10,"msg":"cJZBwcEIh3pqAL","data":
[{"current":2000000,"size":400,"id":300,"createtime":"2023-03-13T23:10:
48.914912209-04:00"},
{"current":3000,"size":200,"id":500,"createtime":"2023-03-13T23:10:48.
914912476-04:00"}],"total":400}

(f) 样例6

图 6　正常样本样例

2.1.3 评价指标

由于系统上线后，执行体的响应数据是否遭受网络攻击是未知的，因此通常利用评价指标准确度（ACC, accuracy）、检出率（DR, detection rate）和误报率（FAR, false alarm rate）来判断响应数据是否异常。ACC 用于衡量整体检测性能，DR 是对正常数据的检测性能，FAR 是对异常数据的误判检测性能，三项指标计算式分别为

$$\mathrm{ACC}=\frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{FP}+\mathrm{FN}+\mathrm{TN}} \tag{2}$$

$$\mathrm{DR}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{3}$$

$$\mathrm{FAR}=\frac{\mathrm{FP}}{\mathrm{FP}+\mathrm{TN}} \tag{4}$$

其中，真正例（TP, true positive）表示正常样本被正确识别出的数量；假正例（FP, false positive）表示异常样本被误报为正常样本的数量；真反例（TN, ture negative）表示异常样本被正确识别的数量；假反例（FN, false negative）表示正常样本被预测为异常样本的数量。

2.1.4 参数设置

本文采用基于 Paddle 的深度学习架构训练自编码−解码的深度学习模型，实验中，Epoch 设置为 200，Batchsize 设置为 128，优化器选用 Adam，学习率设置为 0.001，权重衰减为 0.9，自编码−解码的深度学习模型的阈值设定为 0.02。

2.2 性能对比

2.2.1 检测准确度

为了验证本文方法的有效性，将本文方法与主流的传统拟态裁决方法（全体一致性裁决方法[15]、多数裁决方法[16]、自适应增量式裁决方法[22]和余弦相似度裁决方法[23]）和基于深度学习模型的方法（GANomaly[24]、自编码−解码 0 和自编码−解码 1）进行实验对比，如表 3 所示。其中 GANomaly[24]基于生成对抗网络进行裁决，自编码−解码 0 基于 1 个自编码层和 1 个解码层（阈值设定为 0.032）的方法进行裁决，自编码−解码 1 基于 3 个自编码层和 3 个解码层（阈值设定为 0.016）的方法进行裁决。

由表 3 可知，对比传统的拟态裁决方法，本文方法的性能是最优的，主要是因为传统的拟态裁决方法只是直接对输出的响应数据进行简单对比，没有挖掘不同数据间深度语义特征的相关性，从而导致传统方法的预测效果不佳。通过 ACC 指标可以看出，本文方法的整体检测性能要高于传统的拟态裁决方法。这主要是因为本文方法通过提出的深度学习模型，学习到了软硬件差异导致不同执行体中正常数据不一致的统计规律，大幅度降低了传统的拟态裁决方法中浅层数据特征作为一致性评价标准的敏感度。通过 DR 指标可以看出，本文方法提高了对正常样本的预测准确度，主要是因为深度学习模型能够弱化浅层数据形式上的不一致，从统计分布的角度扩大了正常样本的检测域，从而大幅度减小了假阳现象，提高了正常样本的预测准确度。通过 FAR 指标可以看出，本文方法对异常样本被误判的性能与其他传统的拟态裁决方法的一样。实际上，传统的拟态裁决方法能够检测出异常样本，主要是这些方法采用逐字节比对的方式，很容易区分出数据间不一致，而本文方法则是根据异常样本与正常样本统计分布的偏离度来准确识别异常样本。因此，传统的拟态裁决方法本身存在局限性，一方面，在拟态改造过程中，需要通过逐字节比对的裁决方式标注出不同执行体输出的不一致数据，由于这些不一致数据是软硬件差异化导致的，因此在裁决方法中需要将这些正常情况下的不一致数据全部忽略，方可检测出攻击行为造成的不一致现象；另一方面，由于人工标注引入了人为因素，检测的准确度高度依赖于拟态改造过程中标注出的正常情况下的不一致数据，这就容易造成检测结果的不稳定，影响系统的防御能力。

表 3 不同方法 ACC、DR、FAR 和时延的性能对比

方法	ACC	DR	FAR	时延（180 个样本）/ms
全体一致性裁决	60.00%	5.70%	0	226
多数裁决	81.60%	50.90%	0	182
自适应增量式裁决	82.56%	53.42%	0	468
余弦相似度裁决	78.76%	42.33%	0	238
GANomaly	99.47%	99.02%	0	113
自编码−解码 0	90.38%	85.93%	0	17
自编码−解码 1	99.51%	98.95%	0	34
本文方法	99.50%	99.00%	0	19

此外，本文方法的时延优于传统的拟态裁决方法，主要是因为传统的拟态裁决方法需要通过对不

同的输出响应数据进行逐字节对比或编码来判断它们之间的一致性，对比或编码增加了方法时延，且随着待测样本数的增加，时延也呈线性增长趋势。然而，本文方法则充分发挥了深度学习模型的计算优势，大幅缩短了方法的处理时间。

对比基于深度学习模型的方法，综合考虑方法性能和时延，本文方法较优。当对比的深度学习方法（自编码−解码 0）比本文方法简单时，其预测准确度没有本文方法高，主要是因为过于简化的深度学习模型不利于不同响应数据间深度语义特征的学习；当对比的深度学习方法较为复杂时（GANomaly、自编码−解码 1），虽然这些方法的预测准确度和本文方法近似，但是时延比本文方法高，不利于将复杂方法集成到真实应用的拟态化改造，实现最大化系统服务质量的目的。

因此，本文提出的基于深度学习的拟态裁决方法通过深度学习中自动化的学习机制代替传统的拟态裁决方法需要借助复杂的人为分析方法，全面覆盖了正常情况下不一致数据的潜在语义信息，并准确刻画正常样本的数据分布特征，便于检测异常的攻击行为。而且综合考虑性能和时延两方面因素，本文方法能够在保证较高性能的同时最小化时延，更适用于真实应用的拟态化改造需求。

2.2.2　时延

虽然本文方法的检测准确度高于传统的拟态裁决方法，若本文方法的时延指标较大，则不利于将本文方法应用到拟态化改造的真实应用场景。因此，本节对比了本文方法和多数裁决方法的时延效果，如图 7 所示。由图 7 可知，当测试样本的数量较少时，本文方法的时延与多数裁决方法的时延近似，但随着测试样本数量的增加，本文方法的时延要远小于多数裁决方法的时延。特别地，随着判别样本逐渐增多，多数裁决方法造成的时延是近似线性剧增的，而本文方法则逐渐稳定，主要是因为多数裁决方法是通过不同样本之间字节比对的方式，随着样本的增多，比对时不同执行体产生的样本对的数目也在增加，从而导致时延也逐渐增加；而样本数越多，本文方法越能发挥深度学习模型的计算优势，高速处理待测样本，降低了时延。因此，本文方法更利于应用到真实场景的拟态化改造中，提高目标系统的安全弹性。

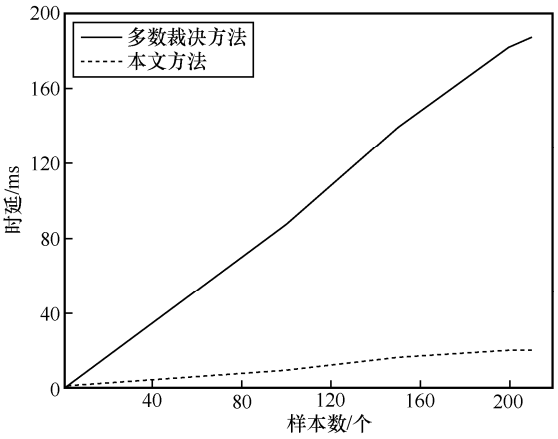

图 7　时延对比

2.2.3　模型优化

由于本文采用了自编码−解码的深度学习模型，为了确定模型的最佳参数，本节进行了如下实验。如表 4 所示，在确保不同自编码−解码深度学习模型参数设置相同的情况下（即与第 2.1.4 节实验参数设置一致），只改变深度学习模型结构获取的预测表现。从表 4 可以看出，最佳的深度学习模型结构为 2 个自编码层和 2 个解码层，能够最大化提取深度语义特征，从而准确学习对应统计规律。通过分析深度网络模型参数的实验（如表 5 所示）可知，当 Epoch 从 50 递增到 200 时，预测性能逐渐增加；当 Epoch 从 200 递增到 250 时，预测性能保持不变，则说明最佳的训练参数是 Epoch 设置为 200，此时本文的深度学习模型趋于稳定状态。

表 4　不同网络结构 ACC、DR 和 FAR 对比

网络结构	ACC	DR	FAR
1 个自编码层和 1 个解码层	87.5%	75.0%	0
2 个自编码层和 2 个解码层	**99.5%**	**99.0%**	0
3 个自编码层和 3 个解码层	97.5%	95.0%	0

表 5　不同 Epoch 下 ACC、DR 和 FAR 对比

Epoch	ACC	DR	FAR
50	90.0%	80.2%	0
100	90.8%	81.6%	0
150	98.0%	95.7%	0
200	**99.5%**	**99.0%**	**0**
250	**99.5%**	**99.0%**	**0**

2.2.4　反馈优化机制测试

为了检验本文提出的反馈优化机制具有增强深度学习模型检测准确度的效果，本节分别对比了

未引入反馈优化机制方法和本文方法，如表 6 所示。由表 6 可知，本文方法（引入反馈优化机制）的预测表现优于未引入反馈优化机制方法的表现，这表明反馈优化机制能够优化深度学习模型，并自适应地提高深度学习模型的裁决准确度。

表 6 反馈优化机制下不同方法 ACC、DR 和 FAR 对比

方法	ACC	DR	FAR
未引入反馈优化机制	98.5%	97.1%	0
本文方法	**99.5%**	**99.0%**	0

2.2.5 真实应用测试

虽然上述的实验结果能够验证本文方法的有效性，但是鉴于仿真数据本身的局限性和可信度，本节将对实际抓取的网页数据进行测试，网页数据来源于某效能考核管理平台（如图 8 所示），通过对整个网页 80 个请求统一资源定位符（URL, uniform resource locator），分别遍历发送请求将近 40 次，获取 3 000 个训练样本和 180 个测试样本（异常数据 90 个、正常数据 90 个），并按照本文方法流程进行测试，测试结果如表 7 所示，其中全体一致性裁决方法[15]和多数裁决方法[16]都是对输出响应数据进行逐字节比对来判别一致性，自适应增量式裁决方法[22]是通过联合字节比对和相似性方法进行一致性判别，余弦相似度裁决方法[23]是余弦相似性进行一致性判别。通过对比可见，在真实应用中，本文方法也优于其他传统的拟态裁决方法，表明深度学习方法增强了对不同响应数据间深度语义特征的理解，准确刻画了它们之间的相关性关系，从而提升了预测的准确度。因此，本文方法在真实应用场景下也能获取较高的检测性能，这表明本文方法在拟态裁决中是有效且可行的。

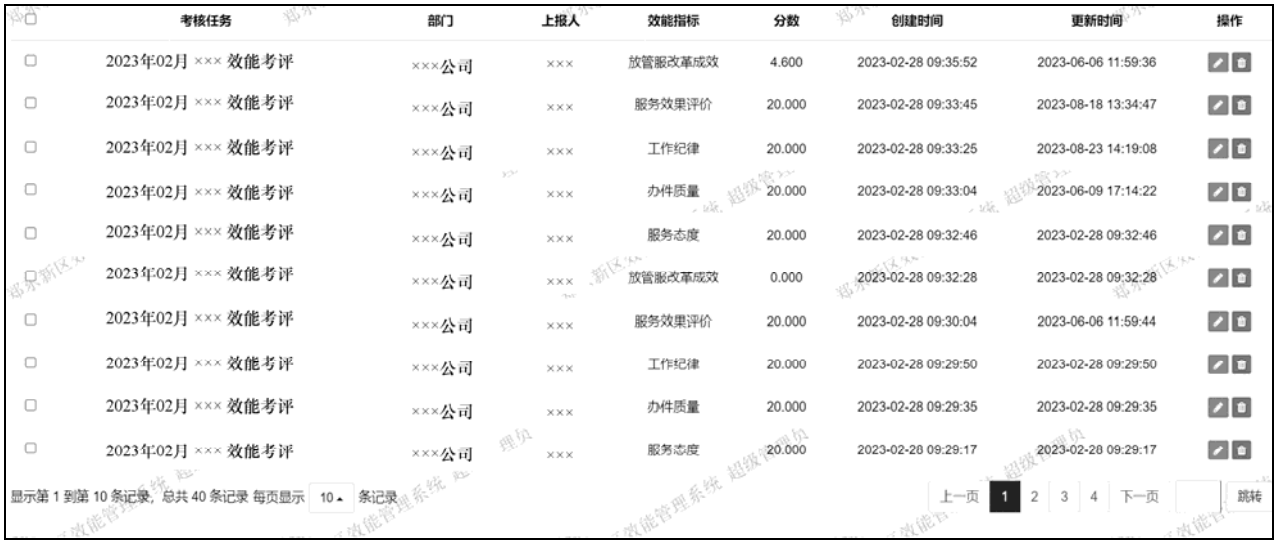

图 8 某效能考核管理平台

表 7 真实场景下不同方法 ACC、DR 和 FAR 对比

方法	ACC	DR	FAR
全体一致性裁决	48.61%	3.27%	22.83%
多数裁决	62.35%	44.21%	19.45%
自适应增量式裁决	65.54%	50.07%	14.56%
余弦相似度裁决	62.13%	48.74%	17.89%
本文方法	**97.20%**	**94.80%**	**3.10%**

3 讨论

3.1 深度学习模型结构

本文选用的自编码−解码的深度学习模型结构比较简单，主要考虑两方面的因素。一方面，较复杂的深度学习模型结构和本文提出的简单深度学习模型结构的预测效果近似相同，综合考虑性能和时延，选择了简单的深度学习模型来满足真实应用拟态化改造的需求。如表 8 所示，本节对比了不同深度学习模型结构下的预测表现，实验设置与第 2.1 节描述的实验设置相同。GANomaly[24]方法采用生成对抗网络结构，其中生成器采用自编码−解码−自编码的网络设计来生成与正常响应数据具有相似统计分布的重塑响应数据，判别器采用分类网络的设计来判别正常响应和生成的重塑响应数据的真假；自编码−解码 1 方法采用三层自编码和三层

解码的网络结构（阈值设定为 0.016）。通过对比实验发现，本文提出的简单深度学习模型和复杂的深度学习模型的预测效果近似相同，但本文方法的时延低于复杂的深度学习方法，这将有利于把本文方法集成到真实应用的拟态化改造来最大化系统的服务质量。另一方面，由于本文方法被应用到了真实业务中，即政府干部效能考核管理平台的拟态化改造，为了最大化整个系统的服务质量，该平台对拟态化改造的响应时延做了约束（平均响应时延不能大于 50 ms）。由于平均响应时延包括网络、缓存、裁决和转发等多因素下的响应时延，因此，在不影响检测性能的条件下，考虑将深度学习模型简单化，可以最小化拟态裁决时延，使其不会因为裁决时延过大而影响整个系统的服务质量。

表 8　不同网络结构下不同方法的 ACC、DR、FAR 和时延对比

方法	ACC	DR	FAR	时延（180 个样本）/ms
GANomaly	99.47%	99.02%	0	113
自编码−解码 1	99.51%	98.95%	0	34
本文方法	99.50%	99.00%	0	19

3.2　阈值合理性分析

本节设置的判断安全威胁的阈值是 0.02，它是通过离线训练阶段损失函数的变化得出的，如图 9 所示。从图 9 可以看出，离线训练阶段，当深度学习模型的损失下降到 0.02 时，深度学习模型达到稳定状态，表明此时的损失能够表征正常响应数据之间的内在规律。因此，在线测试阶段，一旦待测响应输出数据的损失大于该阈值，则表明该响应输出数据偏离了正常响应数据的分布范围，其所在的执行体被认为受到了潜在的安全威胁；反之，若待测响应输出数据的损失小于或等于该阈值，则表明该响应输出数据符合正常响应数据的统计分布，其对应的执行体被认为处于安全状态。

由于不同的阈值会对方法的性能产生不同的影响，为了说明本文方法选取阈值的合理性，本节增加了不同阈值下的性能对比实验，如表 9 所示。从表 9 可以看出，当设定的阈值（0.01 或 0.015）小于本文设置的阈值（0.02）时，ACC 和 DR 值都降低了，这表明部分正响应输出数据被检测为异常，造成了假阳现象，且随着阈值的增加，ACC 和 DR 的值在增加；当设定的阈值（0.03）大于本文设置的阈值（0.02）时，ACC 值降低，FAR 值增加，这表明部分异常响应输出数据被检测为正常，影响方法的预测准确性和异常数据的误判检测度，从而无法精准检测异常数据。因此，综合考虑 ACC、DR 和 FAR 的指标，本文选用阈值为 0.02，此时的阈值不仅能够有助于本文方法获取较高的预测准确度，而且还能防止假阳现象。

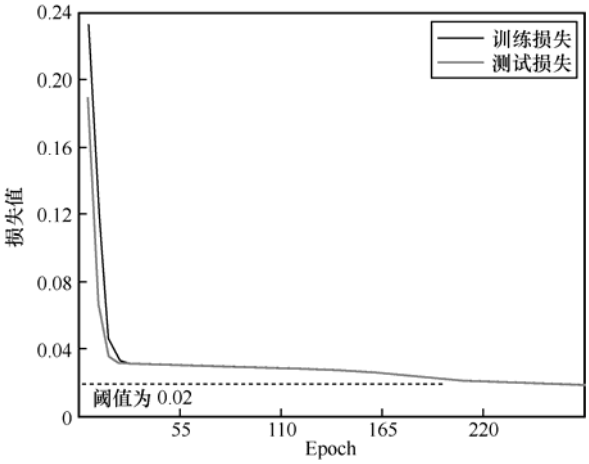

图 9　损失函数的变化曲线

表 9　不同阈值下 ACC、DR 和 FAR 对比

阈值	ACC	DR	FAR
0.01	72.51%	68.35%	0
0.015	83.36%	76.82%	0
0.03	79.84%	98.86%	59.81%
本文阈值 0.02	**99.50%**	**99.00%**	**0**

3.3　深度学习模型输出尺寸

本节选用的自编码−解码的深度学习模型的输出是在综合考虑方法性能和时延条件下，简化自编码−解码深度模型的复杂度，满足真实应用拟态化改造要求设计的。这主要是因为在真实应用拟态化改造中，若裁决方法的性能太低，则达不到检测异常的要求，无法准确识别出安全威胁；若裁决方法的时延太高，则会严重影响系统的正常运行，从而影响系统的服务质量，造成用户的感知体验差。因此，综合考虑性能和时延，通过简化深度学习模型的输出尺寸，能够最小化裁决时延，达到提升整个系统服务质量的目的。

对于 FC_3 输出尺寸是 1×2 的设计，一方面是由于自编码器的重要作用是特征提取，当压缩到简单的 1×2 时能够呈现正常响应数据的信息，这无形中增加了异常响应数据的重建难度[25]，有利于准确检测异常数据；另一方面是基于最简化自

编码–解码深度学习模型结构的原则，在不影响方法性能的情况下最小化裁决的时延，有利于提高拟态改造后真实应用的服务质量。因此，为了验证 FC_3 输出的合理性，本节又对比了 FC_3 层不同输出尺寸下方法预测性能和时延，如表 10 所示。从表 10 可以看出，当 FC_3 层的输出尺寸从 1×20 降低到 1×2 时，方法的性能近似相同，但是随着 FC_3 层输出尺寸的增加，方法的时延在增加；当 FC_3 层输出尺寸降低到 1×1 时，虽然方法时延与输出尺寸 1×2 时相同，但方法的性能却出现明显下降。综上所述，本节选择 FC_3 层输出尺寸为 1×2，它能够降低裁决的时延，使其不会因为裁决时延过大而影响整个系统服务质量的优化。

表 10 FC_3 层不同输出尺寸时 ACC、DR、FAR 和时延对比

FC_3 层输出	ACC	DR	FAR	时延（180 个样本）/ms
1×20	99.50%	98.97%	0	30
1×10	99.48%	99.01%	0	28
1×5	99.50%	99.00%	0	25
1×2	99.50%	99.00%	0	19
1×1	94.23%	91.86%	0	19

4 结束语

针对现有的拟态裁决方法容易将软硬件资源构成的异构执行体输出的正常响应数据的不一致性误判为异常，从而造成假阳现象的问题，本文提出了深度语义特征学习的拟态裁决方法，通过构建简化模型复杂度的自编码–解码深度学习模型，设计联合离线训练和在线裁决的训练机制，使其能够将数据层面的浅层特征转移到深度学习模型的深度语义特征，挖掘正常数据间的相关性关系并归纳其数据间统计分布规律，扩大正常数据的检测域，进而通过在线推演异常数据偏离正常数据统计分布的距离来识别攻击行为，增强系统的安全弹性。此外，为了进一步优化自编码–解码深度学习模型的检测准确度，提出了反馈优化自监督机制，将错误的裁决结果重新送入深度模型进行训练，达到自纠监察深度学习模型准确度的目的。所以，鉴于现有拟态裁决方法的局限性，本文方法能够自适应地学习正常响应数据间的不一致，准确分析异常和正常数据间的差异性，大幅度降低假阳现象，提高拟态防御的效果，为拟态防御的实践应用提供了一种自动化高效维持目标系统安全性的方法。

参考文献：

[1] YANG A M, LU C M, LI J, et al. Application of meta-learning in cyberspace security: a survey[J]. Digital Communications and Networks, 2023, 9(1): 67-78.

[2] WU J X. Development paradigms of cyberspace endogenous safety and security[J]. Science China Information Sciences, 2022, 65(5): 156301.

[3] SEPCZUK M. Dynamic Web application firewall detection supported by cyber mimic defense approach[J]. Journal of Network and Computer Applications, 2023, 213: 103596.

[4] MOLINA-CORONADO B, MORI U, MENDIBURU A, et al. Survey of network intrusion detection methods from the perspective of the knowledge discovery in databases process[J]. IEEE Transactions on Network and Service Management, 2020, 17(4): 2451-2479.

[5] ZHAO R J, GUI G, XUE Z, et al. A novel intrusion detection method based on lightweight neural network for Internet of things[J]. IEEE Internet of Things Journal, 2022, 9(12): 9960-9972.

[6] WU Y H, HU X D. Industrial Internet security protection based on an industrial firewall[C]//Proceedings of the 2021 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA). Piscataway: IEEE Press, 2021: 239-247.

[7] 邬江兴. 网络空间拟态防御研究[J]. 信息安全学报, 2016, 1(4): 1-10.
WU J X. Research on cyber mimic defense[J]. Journal of Cyber Security, 2016, 1(4): 1-10.

[8] 吴铤, 胡程楠, 陈庆南, 等. 基于执行体划分的防御增强型动态异构冗余架构[J]. 通信学报, 2021, 42(3): 122-134.
WU T, HU C N, CHEN Q N, et al. Defense-enhanced dynamic heterogeneous redundancy architecture based on executor partition[J]. Journal on Communications, 2021, 42(3): 122-134.

[9] 周大成, 陈鸿昶, 程国振, 等. 面向持久性连接的自适应拟态表决器设计与实现[J]. 通信学报, 2022, 43(6): 71-84.
ZHOU D C, CHEN H C, CHENG G Z, et al. Design and implementation of adaptive mimic voting device oriented to persistent connection[J]. Journal on Communications, 2022, 43(6): 71-84.

[10] 张进, 葛强, 徐伟海, 等. 拟态路由器 BGP 代理的设计实现与形式化验证[J]. 通信学报, 2023, 44(3): 33-44.
ZHANG J, GE Q, XU W H, et al. Design, implementation and formal verification of BGP proxy for mimic router[J]. Journal on Communications, 2023, 44(3): 33-44.

[11] 姚东, 张铮, 张高斐, 等. 多变体执行安全防御技术研究综述[J]. 信息安全学报, 2020, 5(5): 77-94.
YAO D, ZHANG Z, ZHANG G F, et al. A survey on multi-variant execution security defense technology[J]. Journal of Cyber Security, 2020, 5(5): 77-94.

[12] ZHENG Y, LI Z, XU X L, et al. Dynamic defenses in cyber security: techniques, methods and challenges[J]. Digital Communications and Networks, 2022, 8(4): 422-435.

[13] 邬江兴. 论网络空间内生安全问题及对策[J]. 中国科学: 信息科学, 2022, 52(10): 1929-1937.
WU J X. On endogenous security problems in cyberspace and countermeasures[J]. Scientia Sinica (Informationis), 2022, 52(10): 1929-1937.

[14] 邬江兴. 拟态防御技术构建国家信息网络空间内生安全[J]. 信息通信技术, 2019, 13(6): 4-6.

WU J X. Mimicry defense technology to build endogenous security in national information network space[J]. Information and Communications Technologies, 2019, 13(6): 4-6.

[15] PARHAMI B. Voting algorithms[J]. IEEE Transactions on Reliability, 1994, 43(4): 617-629.

[16] CHOI J, GOH K. Dynamics of consensus formation on multiplex networks: the majority-vote model[C]//Proceedings of APS March Meeting. California: APS Press, 2018: 1-15.

[17] VIHINEN M. Majority vote and other problems when using computational tools[J]. Human Mutation, 2014, 35(8): 912-914.

[18] 武兆琪, 张帆, 郭威, 等. 一种基于执行体异构度的拟态裁决优化方法[J]. 计算机工程, 2020, 46(5): 12-18.
WU Z Q, ZHANG F, GUO W, et al. A mimic arbitration optimization method based on heterogeneous degree of executors[J]. Computer Engineering, 2020, 46(5): 12-18.

[19] 沈丛麒, 陈双喜, 吴春明, 等. 基于信誉度与相异度的自适应拟态控制器研究[J]. 通信学报, 2018, 39(S2): 173-180.
SHEN C Q, CHEN S X, WU C M, et al. Research on adaptive mimicry controller based on credibility and dissimilarity[J]. Journal on Communications, 2018, 39(S2): 173-180.

[20] ZHANG P Y, LI C H, WANG C B. VisCode: embedding information in visualization images using encoder-decoder network[J]. IEEE Transactions on Visualization and Computer Graphics, 2021, 27(2): 326-336.

[21] BECHTLE S, MOLCHANOV A, CHEBOTAR Y, et al. Meta learning via learned loss[C]//Proceedings of the 25th International Conference on Pattern Recognition (ICPR). Piscataway: IEEE Press, 2021: 4161-4168.

[22] 祝永胜, 张铮, 李继忠. 一种 Web 服务用户代理/表决器设计方法[J]. 信息工程大学学报, 2017, 18(5): 613-617.
ZHU Y S, ZHANG Z, LI J Z. Web server user agent/voter design method[J]. Journal of Information Engineering University, 2017, 18(5): 613-617.

[23] 李卫超, 张铮, 王立群, 等. 基于拟态防御架构的多余度裁决建模与风险分析[J]. 信息安全学报, 2018, 3(5): 64-74.
LI W C, ZHANG Z, WANG L Q, et al. The modeling and risk assessment on redundancy adjudication of mimic defense[J]. Journal of Cyber Security, 2018, 3(5): 64-74.

[24] AKCAY S, ATAPOUR-ABARGHOUEI A, BRECKON T P. GANomaly: semi-supervised anomaly detection via adversarial training[C]//Proceedings of Asian Conference on Computer Vision. Berlin: Springer, 2019: 622-637.

[25] YANG G, ZHANG Y, Investigating the relationship between compression ratio, anomaly detection and generative capability of deep auto-encoder networks[J]. Neurcomputing, 2019, 342(1): 202-211.

[作者简介]

杨晓晗（1989− ），女，河南南阳人，博士，信息工程大学助理研究员，主要研究方向为网络空间安全、拟态防御、云计算安全等。

程国振（1986− ），男，山东菏泽人，博士，信息工程大学副教授，主要研究方向为网络空间安全、云计算安全和软件定义网络等。

刘文彦（1985− ），男，河南周口人，博士，信息工程大学助理研究员，主要研究方向为网络空间安全、主动防御等。

张帅（1994− ），男，河南南阳人，博士，信息工程大学助理研究员，主要研究方向为网络空间安全、云计算安全等。

郝兵（1986− ），男，河南开封人，嵩山实验室工程师，主要研究方向为网络空间安全、云计算等。