计 算 机 研 究 与 发 展
Journal of Computer Research and Development

https://crad.ict.ac.cn
62(3): 765−778, 2025

本期推荐

MNDetector: 基于多层网络的异常访问检测方法

袁子淇[1] 孙庆赟[1] 周号益[2] 朱祖坤[1] 李建欣[1]

[1](北京航空航天大学计算机学院 北京 100191)

[2](北京航空航天大学软件学院 北京 100191)

(yuanzq@buaa.edu.cn)

MNDetector: Anomaly Access Detection Method Based on Multiplex Network

Yuan Ziqi[1], Sun Qingyun[1], Zhou Haoyi[2], Zhu Zukun[1], and Li Jianxin[1]

[1] (*School of Computer Science and Engineering*, *Beihang University*, *Beijing* 100191)

[2] (*School of Software*, *Beihang University*, *Beijing* 100191)

Abstract Given the frequent cybersecurity incidents, anomaly detection methods have been widely employed for the identification of malicious behaviors. However, these anomalous accesses often exhibit prominent characteristics only in certain attribute fields, rendering the detection results susceptible to interference from attributes where anomalies are less prominent. To address this issue, MNDetecctor, an anomaly access detection framework that introduces the multiplex network structure into this field is proposed. Through association analysis, closely associated attribute fields are constructed into single-layer networks, with cross-layer connections added to form a multiplex network. Subsequently, cross-layer walks are performed to obtain node sequences within the same layer and across layers, facilitating node embedding. Ultimately, a hierarchical generative adversarial network is employed to merge reconstruction losses and discriminative results across different layers, thereby achieving anomaly access detection. Experimental results demonstrate that MNDetector surpasses the performance of state-of-the-art detection methods on multiple public datasets, achieving an approximately 8% increase in *F*1 score compared with commonly used methods. In-depth case studies elucidate the variation in detection outcomes across diverse scenarios by analyzing the distribution of anomalous attributes within fields. Furthermore, an examination from a network structural perspective clarifies the disparities among results obtained from different layers, substantiating MNDetector's efficacy in addressing the attribute interference issue caused by attribute fields with insignificant anomalous characteristics.

Key words anomaly detection; multiplex network; access detection; network behavior; cyber security

摘 要 针对频发的网络安全事件,异常访问检测被广泛应用于恶意行为的识别.然而,异常访问通常仅在部分属性字段上体现出显著的异常特性,检测结果易被异常特性不显著的字段所干扰.针对这一问题,提出 MNDetector,将多层网络结构引入异常访问检测领域,基于关联紧密的属性字段构建单层网络,并添加层间关联以形成多层网络.随后利用适应多层网络的跨层游走获得同层及跨层节点序列以计算节点表示.最终利用分层生成对抗网络(GAN)融合各层重构损失与判别结果,实现异常检测.实验结果表明,MNDetector 在多个公开数据集上的检测效果超过了最优方法,相较于常用方法实现了约 8% 的 *F*1 分数提

收稿日期: 2023-09-12; **修回日期**: 2024-05-16

基金项目: 国家杰出青年科学基金项目(62225202); 国家自然科学基金青年科学基金项目(62302023); 中国人工智能学会-华为 MindSpore 学术奖励基金

This work was supported by the National Science Foundation for Distinguished Young Scholars (62225202), the National Natural Science Foundation of China for Young Scientists (62302023), and the CAAI-Huawei MindSpore Open Fund.

通信作者: 孙庆赟(sunqy@buaa.edu.cn)

升. 进一步的案例研究通过分析异常特性在属性字段上的分布解释了不同场景的检测效果差异, 并从网络结构的角度解释了各层检测结果差异, 验证了 MNDetector 能够解决异常特性不显著的属性字段造成的属性干扰问题.

关键词 异常检测; 多层网络; 访问检测; 网络行为; 网络安全

中图法分类号 TP183;TP393

DOI: 10.7544/issn1000-1239.202330735 **CSTR:** 32373.14.issn1000-1239.202330735

网络通信逐渐成为现代社会运作的基础设施, 个人、企业及政府机构都依赖网络进行信息和数据的传输, 形成了规模庞大的网络大数据. 随之而来的是日益增长的网络安全威胁, 黑客攻击、数据泄露等一系列恶意事件在物联网[1]、工业互联网[2]等众多领域中都造成了重大损失. 网络安全问题已经成为全球范围内关注的焦点, 对于保护个人隐私、商业机密和国家安全都具有重要意义.

网络安全威胁源于异常的网络访问, 例如链接和传输等. 这些访问的特征和目的随场景变化, 但均与广泛存在的正常访问存在差异, 且在全部访问中占比极低[3]. 现有研究提出了一系列异常访问检测方法, 从访问日志文本、访问行为序列等多个角度进行识别. 其中最主流的思路是基于正常访问数据建立常态模型, 进而评估待测数据与正常数据的相似度, 若低于设定阈值则将其视为异常数据. 在此基础上, 基于图结构的检测方法引入了网络实体与访问之间的复杂关联, 进一步提高了检测效果[4-6].

然而, 异常访问通常仅在部分属性字段中体现出显著的异常特性. 分布式拒绝服务(distributed denial of service, DDoS)攻击在时间方面的属性上呈现出集中的异常特性, 但在传输内容方面的异常特性并不显著; SQL(structured query language)注入攻击在内容方面存在明显的异常特性, 但在传输协议等方面与正常访问差异不大. DDoS 和 SQL 二者体现异常特性的属性层次并不相同[7-8]. 异常特性在属性字段上的分布差异极大地影响了检测效果. 现有的异常访问检测方法忽略了异常特性仅在部分属性字段上显著的问题, 往往将全部属性字段共同嵌入至低维空间, 导致正常属性与异常属性相互混合. 不体现异常特性的属性被引入数据表示结果, 造成了属性干扰问题, 降低了检测效果. 图 1 展示了属性干扰问题对检测结果的影响.

为解决上述问题, 本文提出了 MNDetector, 将多层网络(multiplex network)结构引入异常访问检测领域. 多层网络不仅能通过层次化的网络结构隔离属性字段之间的干扰, 还能建立连接不同层网络的跨层边, 捕捉不同层次间可能存在的隐式关联. 通过基于属性字段集的细粒度划分, 弱化无关字段间的相互影响, 在体现显著异常特性的属性层次进行检测以获得更准确的检测结果. 具体而言, 通过对访问数据进行属性字段的频繁集分析, 利用内部相关性较强的字段集构建单层网络并添加层间关联, 形成多层网络, 实现对字段的分层划分. 在此基础上, 采用适应多层网络结构的跨层游走计算节点嵌入表示, 进而利用分层生成对抗网络(generative adversarial network, GAN)计算未知节点的重构损失与判别结果, 实现异常检测.

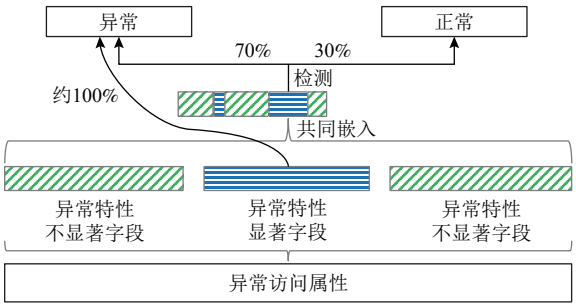


Fig. 1 Schematic diagram of attribute interference problem

图 1 属性干扰问题示意图

本文的主要贡献有 4 个方面:

1)将多层网络结构引入网络异常访问检测领域, 提出了面向网络访问数据的多层网络形式化构建方法, 解决了传统多层网络构建方法依赖人工定义且难以捕获深层关联的问题.

2)形式化地分析了现有检测方法面临的属性干扰问题, 并针对性地提出 MNDetector, 利用多层网络分隔属性字段, 避免异常特性不显著的字段干扰检测.

3)提出了面向多层网络的分层 GAN 机制以及分层检测结果的融合方法, 通过融合各层的重构损失与判别结果, 实现了分层信息的综合检测.

4)对分层检测结果进行了案例研究, 通过不同层网络的结构差异直观地解释了检测结果, 阐释了 MNDetector 能够获得更优效果的原因.

1 相关工作

在真实应用场景中，访问数据往往是离散的，无法直接进行多层网络等复杂结构的构建. 此外，多层网络中的每层网络本质上为图结构，图学习的相关研究与本领域研究有着极大的相关性.

1.1 多层网络的构建与学习

多层网络是特殊的异构图，其在多个不同的视角或维度上划分数据样本，分别构建单层网络并在其间添加跨层连边. 多层网络的具体分层逻辑与方法并没有通行标准，在不同领域的应用中可以基于时空尺度、语义层级、先验知识等多种角度构建不同的多层网络，从而面向下游任务实现针对性的数据划分.

与多层网络相近的常用概念有 3 个：多维网络（multi-dimensional network）、分层网络（multi-network）和多视角图（multi-view graph）. 其中，多维网络的概念与多层网络相同，存在二者混用的情况. 分层网络更为常见，即将原始数据构建为多个单层网络结构，但层与层之间没有直接的关联. 例如不同区域的网络拓扑、不同网络平台的用户关联等. 多视角图可以被视为多层网络的子集，其通过每层网络表示单一视角下的数据关联，如社交网络中的关注关系、好友关系、评论关系等. 多视角图中的跨层边往往表示不同视角间的关联，从而描述多个视角的交互模式.

多层网络并未直接划分数据集合，而是从属性字段集的视角出发对数据属进行划分，跨层边则体现了属性字段间的关联紧密程度. 因此其与上述概念均有所差异.

1.1.1 多层网络的构建

将原始数据转换为图结构是多层网络构建的第 1 步，其主要思想是利用节点表示原始数据中的每个样本，将数据特征转换为节点属性，将样本之间的关联转换为连接节点的边，通过边权重表示样本间的相似程度. Chu 等人[9] 提出了上述构建过程的核心思路，对于给定的节点组，利用高斯函数计算节点 u 与节点 v 之间的边权重 w_{uv}，其核心计算过程为

$$w_{uv}=\exp\left(\frac{-\|\boldsymbol{h}_u-\boldsymbol{h}_v\|_2^k}{2\varepsilon^2}\right), \tag{1}$$

其中 $\boldsymbol{h}_u$ 与 $\boldsymbol{h}_v$ 为节点 u 与节点 v 的属性向量，ε 与 k 均为人为指定的参数.

在此基础上，把若干单层网络连接成多层网络的方法包括人为定义和基于关联进行计算.

人为定义的方法主要依赖人工特征，例如通过单层网络之间的相似度设置层间关联的强弱[10]，常用指标有余弦相似度、Jaccard 相似度和 Dice 系数等. 若 2 层网络之间的相似度超过某个阈值，则为其添加跨层边. 这一思路的典型代表是 Park 等人[11] 提出的 DMGI，其定义了属性化的多层网络并将其应用于后续学习任务中.

基于关联的方法则利用应用场景中的关联关系构建跨层边，例如针对不同社交平台用户数据构建的多层网络，在表示相同账户使用人的节点之间添加连边等，因此其应用往往局限于具体场景.

1.1.2 多层网络的学习

Liu 等人[12] 总结了 3 类多层网络学习的实现方法.

第 1 类是把多层网络简单叠加，聚合每层网络中表示相同实体的节点，同时合并重合的边，即把多层网络转换为常规的单层网络，进而直接应用成熟的图学习方法. 这种方法可以直接复用现有的图表示学习方法，但效果不佳.

第 2 类是分别对每层网络进行嵌入，即针对每层网络复用现有的节点嵌入方法，最终将全部嵌入结果进行简单合并. 其本质是把多层网络看作彼此分离的单层网络，忽略了单层网络之间的关联关系.

第 3 类是能够进行跨层学习的方法. 相比于前 2 类方法，其采用的随机游走等机制能在学习过程中直接引入跨层结构信息[13-14].

除此之外，由于网络结构的相似性，部分面向多视角图的异常检测方法也可迁移应用至多层网络检测. 最具代表性的工作是 ANOMULY[15]，其侧重不同场景下的边结构，将分层计算的节点嵌入作为层次化的节点状态，利用门控循环单元（gate recurrent unit，GRU）捕获网络的时间特性以实现动态多层网络的异常检测. 然而其并未针对性设置多层网络节点间的跨层关联，且未具体分析属性字段，导致检测效果难以进一步上升. 类似地，AnomMAN[16] 将不同层次网络的结构与属性嵌入到相同的子空间中，利用注意力机制融合嵌入结果，进而通过重构损失计算对应节点的异常分数.

总的来说，现有研究大多停留在表征学习的领域中，与具体应用场景结合不紧密，难以应用至真实的异常检测任务. 且现有多层网络构建方法大多完全依赖人为定义，可迁移性较差. 这些都为多层网络结构在异常检测领域的应用带来了阻碍.

1.2 图学习

图学习的目标是获取图结构的有效表征以支撑下游任务. 近年来, 深度学习技术迅速发展, 图深度学习方法被不断提出, 正在逐步取代经典方法.

1.2.1 经典图学习方法

最经典的图学习方法是图信号处理(graph signal processing, GSP), 其核心思路是将传统信号处理的方法迁移至图结构中[17]. GSP 利用邻接矩阵、拉普拉斯矩阵等数据结构表示图, 使用矩阵乘法实现滤波器的放缩作用. 这种方式是传统信号处理方法的拓展, 不能对图进行有效的分解, 且时间开销较大, 仅在少数特定领域获得了应用.

在此基础上, 研究者针对图结构的特点提出了矩阵分解方法, 即对特征矩阵进行分解, 在得到的低维矩阵中保留原始特征信息, 将其作为图学习的结果. 基于矩阵分解的图学习可以分为基于拉普拉斯矩阵和基于邻接矩阵两大类, 但由于这 2 类方法都需要存取大量矩阵, 内存消耗较大, 在应用中存在局限.

基于随机游走的图学习方法是应用较广的经典方法. 其通过节点之间的随机游走路径反映关联关系, 其中影响力最大的方法是 DeepWalk[18] 和 node2vec[13]. 在此基础上, Chanpuriya 等人[19]、Zhai 等人[20] 针对社交、推荐等多个场景进一步发展了相关方法. 随机游走得到的游走路径和图结构之间存在紧密关联, 在稀疏网络中表现优越. 但存在依赖随机概率、路径聚合方式难以选择等问题, 需要根据应用场景进行细致调整.

1.2.2 图深度学习方法

图像处理等领域中的深度学习技术被不断迁移至图学习领域. 其代表是 SDNE[21] 及其改进与应用[22]、图卷积网络(graph convolutional network, GCN)[23] 及其优化方法[24] 等. 此外, 针对拓扑不均衡[25]、子图结构[26] 等重要问题也存在相应的学习方法研究. 其中, 以 GCN 为代表的卷积方法成为了图学习的主流. 此类方法将特征矩阵作为输入, 利用消息传递机制进行属性聚合, 通过多轮迭代实现节点嵌入. 这种思路的进一步发展以 SpecAE[27] 为代表, 其加入了反卷积层, 使用高斯混合模型实现图学习. 注意力[28]、动态化[29]、信息瓶颈[30] 等机制与理论的引入提高了图学习的有效性和稳定性.

另一种图深度学习的实现方法是将自编码器的思想与图神经网络相结合, 变分图自编码器是此类方法的典型[31]. 其将邻接矩阵转换为含有节点属性信息的嵌入, 同时使用矩阵的数量积进行解码, 将解码后的邻接矩阵与原始邻接矩阵之间的交叉熵作为整体模型的损失函数. 在此基础上, GraphGAN[32] 结合了随机游走和生成对抗机制, 将图转化为最小生成树, 从根节点开始游走, 通过改变根节点来获得不同路径, 进而通过最小化重构损失实现表征学习.

图深度学习方法避免了经典方法需要手动配置大量参数的问题, 具有良好的普适性. 但由于现有模型普遍使用梯度下降等方法进行参数优化, 其存在与神经网络相同的易陷入局部最优解、易发生梯度消失等问题. 总的来说, 现有的图学习方法大多专注于单层结构, 无法有效区分不同视角、不同属性字段并无法捕获其间的关联关系, 因此无法直接应用于多层网络的学习.

1.3 异常检测

异常检测任务涉及到的场景非常丰富, 且数据特征不同, 需要根据具体场景选择相应的检测方法. 在现有方法中, 部分研究直接将分类方法迁移应用至异常检测领域. 除此之外, 针对异常数据自身的研究可以大体分为 3 类: 基于统计的方法、基于聚类的方法和基于残差分析的方法[33].

1.3.1 基于统计的方法

基于统计的异常检测方法假定数据符合某种特定的统计分布, 从而可对全体样本进行拟合以得到正常数据分布, 不符合该分布的数据即被视为异常数据.

此类方法中最具代表性的是以扫描统计量作为异常指标的检测方法, Kulldorff 统计量[34]、Poisson 统计量[35] 等一系列指标被逐步提出. Wang 等人[36] 进一步通过校准的非参数扫描统计量实现了真实场景下异常子图的有效检测.

统计方法是最基础的异常检测方法, 在特定领域中有一定应用, 但存在泛化性不足的问题.

1.3.2 基于聚类的方法

异常检测任务最基本的特性是数据标签的高偏分布, 即绝大多数数据样本均为正常. 针对这种特殊性质, 聚类方法被大量应用. 其核心思想是识别远离样本中心的少量样本并将其视为异常. 其中, 最常用的是基于欧氏距离的聚类及其改进方法[37], 在特征空间、嵌入空间等选定的空间中, 距离更近的样本被认为更可能从属同一类别. 基于这一思路的 NetWalk[38] 使用基于欧氏距离的聚类方法进行异常检测, 在多个场景中获得了较高的准确率.

1.3.3 基于残差分析的方法

基于残差分析的方法将原始数据编码后进行重

构，由于正常数据占比远大于异常数据，因此重构结果趋向于正常数据模式，从而可以将原始数据与重构数据之间的重构损失作为异常指标.

深度自编码器[39]是此类方法的代表，其通过深度学习拟合正常数据模式以实现非监督的异常检测. 在此基础上，Ding 等人[40]提出的 AEGIS 模型通过图差分编码网络构建编码器与解码器、计算待测节点的重构损失，获得了较高的准确率. Fan 等人[41]进一步引入了图正则化项，保持了图数据局部结构的一致性，减少了检测时间并提高了鲁棒性.

总而言之，现有异常检测方法面对不同场景需要进行针对性的调整，迁移性较差，且大多仅作为表征学习的下游任务，与数据表征方法结合不紧密.

2 问题定义

多层网络可被形式化地定义为图结构 $Graph=\{N_l, E_l, E_{\text{cross}}\}_{l=1,2,\cdots,m}$，其中 N_l 为第 l 层网络的节点集合，E_l 为其对应的边集合，E_{cross} 为跨层边集合，m 为网络层数. 进而异常访问检测可被转化为多层网络上的异常节点检测问题，即寻找特定的节点集合，其中节点的属性与其余节点显著不同.

各类基于图神经网络的检测方法推动了检测效果的提升，为多层网络异常检测提供了有效工具，但同时带来了严重的属性干扰问题.

2.1 图异常检测的属性干扰

以 AnomMAN[16]和 AnomalyDAE[42]等被广泛运用的基于重构的检测方法为例，当处理节点属性时，均方误差等损失函数会使相邻节点的属性表征趋同；当处理边结构时，迭代训练在使损失趋于零的同时会使得重构结构与原始结构相同，从而边重构方法同样会导致相邻节点的属性表征趋同.

在此过程中，节点 u 的属性表征可表示为

$$\boldsymbol{h}_u^{(t+1)}=\sigma\left(\sum_{n\in Ne_s(u)} c_n\boldsymbol{h}_n^{(t)}\boldsymbol{W}^{(t)}\right), \tag{2}$$

其中 $\boldsymbol{h}_n^{(t)}$ 为节点 n 在第 t 次迭代后的属性表征向量，c_n 为对应的权重，$\boldsymbol{W}^{(t)}$ 为可训练参数，σ 为可选择的激活函数，$Ne_s(u)$ 为节点 u 的 s 阶之内的邻居节点集合.

对于相邻节点 u 与 v，当训练使得损失趋于零时，其表征向量趋于相等. 由于在 2 次迭代后节点 u 可捕获到其 $2s$ 阶之内的邻居信息，因此不失一般性地，可以使用相邻节点表征的关系作为训练结束的标志，即 $\boldsymbol{h}_u=\boldsymbol{h}_v$. 从而在每次迭代中，模型将趋向于

$$c_u\left(\sum_{n\in Ne_s(u)}\boldsymbol{h}_n^{(t)}\boldsymbol{W}^{(t)}\right)=c_v\left(\sum_{n\in Ne_s(v)}\boldsymbol{h}_n^{(t)}\boldsymbol{W}^{(t)}\right). \tag{3}$$

针对每次具体的迭代过程，将 $\boldsymbol{h}_n$ 按维度展开，考察第 i 个字段，可将式(3)转写为

$$c_u\boldsymbol{W}_i\left(\sum_{n\in Ne_s(u)}h_{ni}\right)=c_v\boldsymbol{W}_i\left(\sum_{n\in Ne_s(v)}h_{ni}\right), \tag{4}$$

其中 $\boldsymbol{W}_i$ 为模型参数矩阵的第 i 列，h_{ni} 为节点 n 的第 i 维属性值.

对于固定的节点 u 与节点 v，当 u 和 v 的邻居节点属性之和存在较大偏差时，模型训练会使得 $\boldsymbol{W}_i$ 变小，乃至趋近向量 $\boldsymbol{0}$，从而降低该属性字段的影响. 而当属性之和偏差较小时，$\boldsymbol{W}_i$ 则会远离向量 $\boldsymbol{0}$，从而强化该字段的影响.

一般而言，对于能够体现异常特性的属性字段，由于取值与异常与否呈强相关，字段上各节点的属性和偏差较大，而不体现异常特性的属性字段则反之. 这使得图学习过程强化了不体现异常特性的属性字段的影响，抑制了异常特性显著的属性字段的影响，降低检测效果，造成属性干扰问题.

2.2 属性干扰的表现与影响

针对 UNSW-NB15 数据集[43]所构建的溯源图，考察每个属性字段对异常标签的区分度及其邻居节点属性和的波动性之间的关联. 该数据集的详细介绍如 4.1.1 节所示.

具体地，以每个字段对异常标签的信息增益代表其对异常的区分能力，使用变异系数作为取值波动性的评估标准. 图 2 展示了每个字段上信息增益与变异系数的关系，为保证展示效果清晰，信息增益低于 0.1 的字段被去除.

图 2 中，DoS 与 Generic 是 2 类不同的异常访问. 从图 2 中可以看出，变异系数相差较大的属性字段

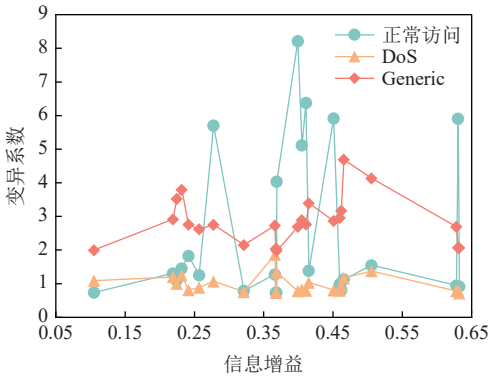


Fig. 2 Association between volatility and information gain of attribute fields

图 2 属性字段波动性与信息增益的关联

信息增益也较高，这意味着对异常检测提供更多信息的字段被抑制，异常特性不显著的字段则产生了更大的影响，即属性干扰问题的存在对检测产生了负面影响.

另一方面，针对不同类型的异常访问，具有异常区分度的属性字段并不完全一致. 这意味着对各类异常进行统一检测时，无法准确定位最具区分度的属性字段，从而影响了综合检测效果.

进一步地，选取信息增益最高的字段，设置异常数据占比为 10%，以 $F1$ 分数作为综合检测指标，将离散值展开成为独热码，使用最基础的 GCN 进行异常检测. 随后，基于信息增益从高至低地增加使用的字段并更新检测结果. 图 3 展示了对应的 $F1$ 分数变化趋势.

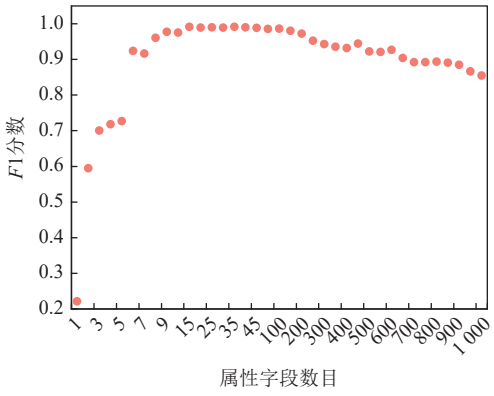


Fig. 3 Association between $F1$ scores and the numbers of attribute fields

图 3 $F1$ 分数与属性字段数目的关联

由图 3 可见，仅采用若干具有更高信息增益的属性字段即可达到良好的检测效果，在使用约 15 个字段时即可达到最优. 随着独热码带来的新字段的引入，检测效果有所降低. 这种降低即不体现异常特性的属性字段干扰综合检测的直观效果.

3 MNDetector 的框架设计

为解决属性干扰对综合检测的影响，本文提出了 MNDetector. 图 4 展示了 MNDetector 的框架，可分为 3 个主要环节：首先构建多层网络结构，随后对多层网络进行学习来获得节点的表征向量，最终使用分层的 GAN 实现异常检测.

3.1 基于关联的多层网络构建

利用多层网络分隔不同语义层次的属性字段可以很好地屏蔽字段间的相互影响. 为了利用多层网络实现异常访问检测，首先需要基于访问数据构建多层网络. 图 5 展示了多层网络构建的过程. 针对原始数据中的字段，利用关联分析获取属性字段频繁集，进而基于相关性较强的字段构建单层网络，最后添加层间关联.

3.1.1 频繁集构建

为了获取属性字段之间存在的隐含关系，通过逐层搜索的迭代方法计算数据属性中的项集关系[44]. 在此基础上，计算属性字段之间的关联规则.

具体而言，在将离散值转换为独热码并删除无效值后，计算每个字段 a 在全部数据中的出现概率 $P(a)$：

$$P(a)=\frac{count(a)}{\sum_{field\in F} count(field)}, \tag{5}$$

其中 F 为全部属性字段的集合，函数 $count$ 表示字段非空的数据条目数.

将所有 $P(a)$ 不为 0 的字段合并为候选集，在其中剔除概率过小的字段即可组成频繁项集. 将单一

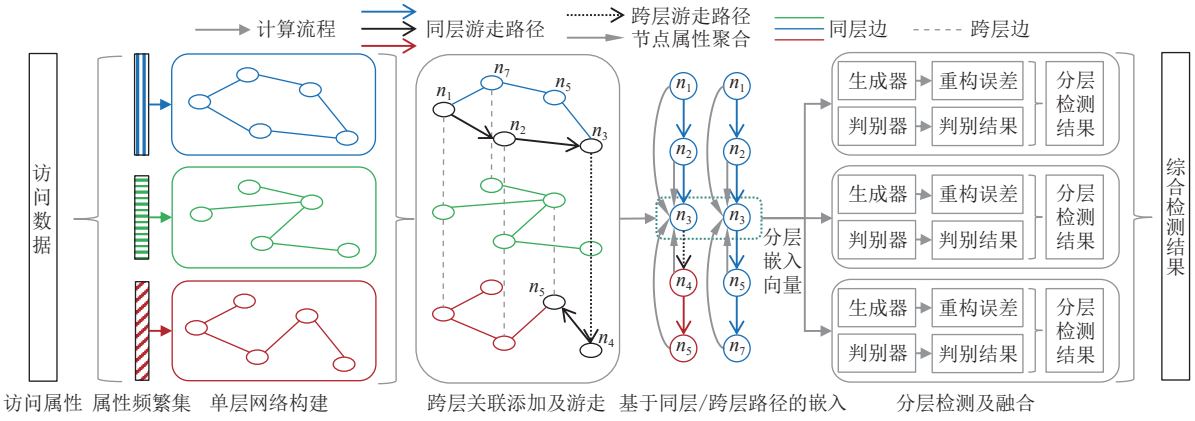


Fig. 4 Overview of MNDetector

图 4 MNDetector 概览

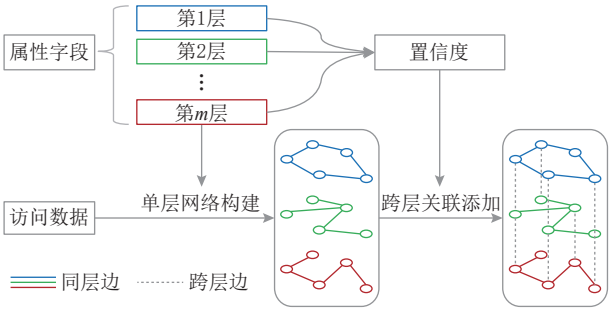


Fig. 5 Multiplex network construction base on association

图 5 基于关联的多层网络构建

字段的统计频率替换成频繁项集的统计频率，通过多轮迭代即可获得全部的频繁项集. 其去重后对应的字段集即为构建每层网络的基础.

3.1.2 网络构建

在每层网络中，节点代表访问行为，节点属性代表访问行为在对应属性字段上的值. 进而利用 L2 范数计算边权重，基于高斯距离形成亲和矩阵，完成对单层网络的构建.

具体地，对于第 l 层上的节点 u 与节点 v，基于节点属性通过式(6)计算同层节点间的边权重.

$$w_{uv}^{(l)} = \exp\left(\frac{-\left\|\boldsymbol{h}_u^{(l)} - \boldsymbol{h}_v^{(l)}\right\|_2^2}{2\varepsilon^2}\right). \tag{6}$$

在获取全部的单层网络之后，通过通用的基于条件概率的置信度添加跨层关联. 也就是，针对任意 2 层网络，将层间置信度作为层间关联的构建概率，从而遍历全部节点即可完成跨层边的添加.

3.2 基于跨层游走的多层网络节点表示

图 6 展示了多层网络的节点表示方法. 基于所构建的多层网络，采用跨层有偏随机游走的方法获取同层节点序列和跨层节点序列，利用全部的节点序列计算中心节点表示.

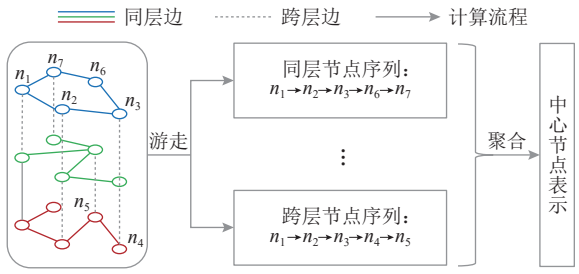


Fig. 6 Multiplex network node representation based on cross-layer walks

图 6 基于跨层游走的多层网络节点表示

在每层网络中，利用邻接矩阵表示网络的结构信息，当节点 u 与节点 v 之间存在边时，基于式(7)计算从 u 到 v 的游走概率.

$$P(v|u) = \frac{\alpha_{pq} w_{uv}}{c}, \tag{7}$$

其中 c 为归一化常数. 参数 α_{pq} 由节点关系以及参数 p 和 q 决定，若节点处于同一层且属性完全一致，则取其为 p 的倒数；若节点处于同一层且互为邻居，则取其为 1；若节点处于不同层且由跨层边相连，则取其为 q 的倒数.

显然，参数 p 和 q 可调整随机游走的偏向性. 当 p 较小时，游走倾向于在同一层上进行，中心节点表示中包含了更多的同层结构信息；当 q 较小时，随机游走倾向于在不同层上进行，中心节点表示中包含了更多的跨层结构信息. 不妨令 u 为游走路径的中心节点，可利用式(8)基于获取到的全部路径计算中心节点的表示.

$$\boldsymbol{h}_u = \underset{path}{mean}\left(\frac{1-\gamma}{len-1}\left(\sum_{n\in N_{path}, n\neq u} \boldsymbol{h}_n\right) + \gamma \boldsymbol{h}_{u,\text{raw}}\right), \tag{8}$$

其中，$\boldsymbol{h}_{u,\text{raw}}$ 表示节点 u 的原始属性，$path$ 表示游走路径，len 表示路径长度，N_{path} 表示路径中的节点集合，γ 表示中心节点权重.

由于网络层次划分的依据是属性字段集的关联，因此相较于同层路径，处理跨层路径时应以更低的权重引入跨层信息.

3.3 基于分层 GAN 的异常检测

多层网络具有分层结构，因此需针对每层网络分别设置 GAN，利用不同层次对应的属性进行训练.

图 7 展示了具体的检测方法. 在训练良好的生成器与判别器后，即可使用生成器获取未知节点的重构损失，使用判别器获取未知节点的判别结果，将生成器和判别器二者融合获得层次化的检测结果，最终将各层检测结果融合成为综合检测结果.

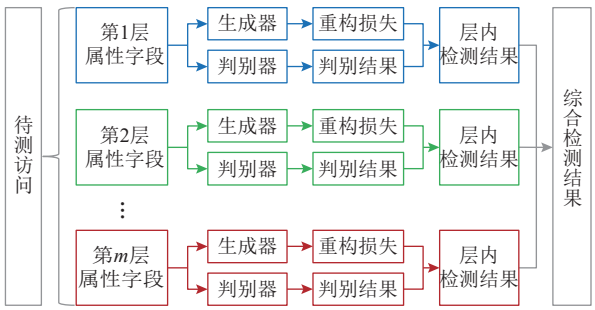


Fig. 7 Anomaly detection based on hierarchical GANs

图 7 基于分层生成对抗网络的异常检测

具体地，利用层内的节点进行训练以获得适应不同层数据分布的生成器与判别器，每个单独的

GAN 的总体损失为

$$loss = E_{x\sim P_{\text{data}}(x)}(\text{lb}D(\boldsymbol{x})) + E_{z\sim P_{\text{noise}}(z)}(\text{lb}(1-D(G(\boldsymbol{z})))), \quad (9)$$

其中 G 代表生成器，D 代表判别器，$\boldsymbol{x}$ 代表输入数据向量，$\boldsymbol{z}$ 代表噪声向量，$P_{\text{data}}(\boldsymbol{x})$ 代表数据自身的分布，$P_{\text{noise}}(\boldsymbol{z})$代表噪声分布.

在各层训练结束后，首先将待测节点的表征向量 $\boldsymbol{h}$ 作为判别器的输入，得到判别器的判别结果 Res_D. 同时将 $\boldsymbol{h}$ 作为生成器的输入，得到生成器的生成结果 $\boldsymbol{h}_G$，将 $\boldsymbol{h}$ 和 $\boldsymbol{h}_G$ 之间的欧氏距离作为生成器的重构损失. 由于不同层的属性数量和属性值有所差异，不同层上节点的重构损失数值无法直接进行比较. 因此需将重构损失归一化，记为 Res_G，进而利用式(10)计算最终的异常检测结果 Res.

$$Res = \beta \times Res_D + (1-\beta)Res_G, \quad (10)$$

其中参数 β 可控制对结构信息或属性信息的侧重程度.

在一些层次的网络中，部分数据样本并未体现出有效差异，即其在对应的属性字段上的值几乎完全一致，这导致图中存在近乎全连接的庞大子图. 这种情况意味着本层网络并未体现出异常区分度，因此相关层次检测结果不被用于最终检测结果的融合.

3.4 复杂度分析

MNDetector 将传统单层图扩展为多层网络，相较于常规的图异常检测方法，其时间、空间开销均有所上升.

记 N 为单层网络节点数目，E 为未形成近乎全连接图的层次中最大的边数目，m 为网络层数，len 为游走路径长度. 游走与嵌入阶段的时间复杂度可估算为 $O(len\times mN)=m\times O(len\times N)$，即单层方法的 m 倍. 由于各层字段的并集即为全部字段，空间复杂度的上升仅体现为更多路径的储存消耗，从而同样为单层检测方法的 m 倍. 在检测阶段，针对单层网络构建的 GAN 的时空开销与单层图方法无异，而空间复杂度来源于 GAN 的储存需求，二者均为单层方法的 m 倍.

综上所述，MNDetector 相较于常规的检测方法，其时间、空间复杂度均有所提高，其扩大倍数为多层网络层数. 考虑到具体场景中所构建的多层网络往往低于 10 层，因此二者没有数量级上的显著差异.

4 实验结果与分析

4.1 实验设置

4.1.1 数据集

采用了 4 个公开数据集进行实验.

1）KDD[45]（KDD CUP 99）. 该数据集模拟了完整的局域网系统，采用不同的攻击方式对虚拟系统进行攻击，汇总了网络连接流量数据及其特征.

2）NSL[45]（NSL-KDD）. 该数据集在 KDD CUP 99 数据集的基础上进行精简，更简要地描述了网络环境.

3）UNSW[43]（UNSW-NB15）. 该数据集通过模拟网络攻击并对网络数据包进行捕获，经过筛选后组成了包含 9 种攻击类型的网络访问记录.

4）TOR[46]（ISCXTor2016）. 该数据集包括描述 Tor（the onion router）网络连接的各方面特征，如连接建立、数据传输等，常用于研究和分析匿名网络.

表 1 展示了上述数据集的详细参数.

Table 1 Parameters of Datasets

表 1 数据集参数

数据集	特征数	异常类别数
KDD	41	39
NSL	42	39
UNSW	49	9
TOR	29	1

4.1.2 基线方法

采用了 6 种基线方法.

1）OC-SVM[47]. 传统异常检测方法的代表，其基本思想是构建最优分类超平面，从而划分正常节点与异常节点. 该方法的检测效果可以作为任务难易程度的判断依据.

2）DAGMM[48]. 基于自编码器的异常检测方法的代表，其首先通过自编码器压缩网络，获取数据表示结果并输入估计网络，得到最终的检测结果.

3）AnoGAN[49]. 基于 GAN 的传统异常检测方法的代表，其学习正样本的潜在分布模式，依据判别结果进行反向传播优化，将判别器的输出作为异常值.

4）BiGAN[50]. 基于 GAN 的图异常检测方法的代表. 其学习额外的编码器，在最终异常检测的同时将生成器和编码器的输出作为判别器的输入，以此来获得更加准确的检测结果.

5）AnomMAN[16]. 基于多视角图的异常检测方法的代表，是当前针对层次化图结构进行检测的最优方法. 其针对不同层次的图分别进行属性矩阵与邻接矩阵的嵌入，进而计算各节点的异常分数.

6）AnomalyDAE[42]. 基于图自编码器的异常节点检测方法的代表. 其分别针对属性矩阵与邻接矩阵进行编码与解码，融合重构损失以实现异常检测.

OC-SVM 作为传统方法，参数量显著低于 MNDe-

tector 和其余的基线方法. 此外, 由于嵌入、检测模型的参数往往与输入属性维度呈正比, 因此多层网络结构的引入不会改变参数量的数量级. 在多层网络检测所独有的结果融合阶段不需要学习, 仅需要引入若干超参数, 不会影响模型总体参数量. 因此, 除 OC-SVM 外的基线方法的参数量与 MNDetector 的参数量保持了一致的数量级.

4.1.3 参数设置

由于原始数据中异常数据占比较高, 为模拟真实的异常检测应用场景, 本文对原始数据进行采样, 使得数据中异常数据占比为 5%.

记单个 GAN 对应的属性空间维度为 *dim*, MNDetector 所使用的生成器由 *dim*×512 的全连接层、ReLU 激活层以及 512×*dim* 的全连接层组成. 其对应的判别器由 4 层神经网络组成, 分别为 *dim*×512 的全连接层、ReLU 激活层、512×1 的全连接层以及 sigmoid 激活层, 全局训练轮次为 10 000. 在计算边权重时 ε=0.1 以降低距离较远的节点之间的影响, 式(7)中 p=1, q=0.8, 以促使模型更多地获取跨层信息. 为处理同层、跨层路径信息重要性不同的问题, 对于式(8)中的参数, 针对同层路径, γ=0.5; 针对跨层路径, γ=0.8. 由于跨层随机游走涉及多个网络层, 采用的属性字段为跨层路径中心节点所在层的字段. 对式(10)中的参数, β=0.5, 以平衡结构信息和属性信息.

4.1.4 评价指标

采用准确率(*accuracy*)、精确率(*precision*)、召回率(*recall*)、*F*1 分数(*F*1_*score*)对实验结果的性能进行评价. 记检测结果真阳性数为 *TP*、假阳性数为 *FP*、真阴性数为 *TN*、假阴性数为 *FN*, 从而

$$accuracy=\frac{TP+TN}{TP+TN+FP+FN}, \tag{11}$$

$$precision=\frac{TP}{TP+FP}, \tag{12}$$

$$recall=\frac{TP}{TP+FN}, \tag{13}$$

$$F1_score=\frac{2\times precision\times recall}{precision+recall}. \tag{14}$$

4.2 综合检测

在 4 个数据集上对 MNDetector 与基线方法的检测效果进行对比, 表 2 展示了相应的综合检测结果.

Table 2 Comprehensive Detection Results on Various Datasets

表 2 各数据集上的综合检测结果

模型	KDD				NSL			
	准确率	召回率	精确率	*F*1 分数	准确率	召回率	精确率	*F*1 分数
OC-SVM[47]	0.978 1	0.770 4	0.787 9	0.779 1	0.979 6	0.763 0	0.817 5	0.789 3
DAGMM[48]	0.987 0	0.881 5	0.862 3	0.871 8	0.990 4	0.933 3	0.881 1	0.906 4
AnoGAN[49]	0.994 4	0.940 7	0.947 8	0.944 2	0.994 4	0.937 0	0.951 1	0.944 0
BiGAN[50]	0.994 0	0.922 2	0.957 0	0.939 3	0.992 2	0.909 6	0.932 4	0.920 9
AnomMAN[16]	0.995 9	0.954 1	0.964 4	0.959 2	0.995 6	0.951 9	0.959 0	0.955 4
AnomalyDAE[42]	0.993 0	0.964 1	0.902 9	0.932 5	0.996 2	0.977 8	0.948 5	0.962 9
MNDetector（本文）	**0.998 0**	**0.979 3**	**0.980 2**	**0.979 7**	**0.998 1**	**0.981 7**	**0.980 8**	**0.981 2**

模型	TOR				UNSW			
	准确率	召回率	精确率	*F*1 分数	准确率	召回率	精确率	*F*1 分数
OC-SVM[47]	0.978 5	0.770 4	0.793 9	0.782 0	0.970 8	0.658 3	0.731 5	0.693 0
DAGMM[48]	0.989 1	0.888 9	0.892 9	0.890 9	0.990 6	0.920 8	0.894 7	0.907 6
AnoGAN[49]	0.989 1	0.874 1	0.905 6	0.889 6	0.993 8	0.945 4	0.931 1	0.938 2
BiGAN[50]	0.993 9	0.931 1	0.945 5	0.938 2	0.992 9	0.945 8	0.915 3	0.930 3
AnomMAN[16]	0.996 2	0.954 4	0.969 9	0.962 1	0.996 4	0.953 3	**0.974 4**	0.963 7
AnomalyDAE[42]	0.996 1	0.963 0	0.959 4	0.961 2	0.990 8	**0.966 7**	0.865 7	0.913 4
MNDetector（本文）	**0.997 5**	**0.974 8**	**0.975 5**	**0.975 1**	**0.996 8**	0.962 5	0.973 9	**0.968 2**

注：黑体数值表示对应指标下的最优结果.

4.2.1 KDD, NSL, TOR 的检测结果

在 KDD, NSL 与 TOR 数据集上, MNDetector 在全部指标上都超过了基线方法, 相较于基线方法的平均效果, MNDetector 实现了约 8 个百分点的提升.

为了进一步分析 MNDetector 的有效性, 在 KDD 数据集中选取了表 3 所示的 2 个样本进行案例研究. 该数据集被构建为 4 层的网络.

Table 3 Information of the Selected Samples in KDD Dataset

表 3 KDD 数据集中所选取样本的信息

属性字段或信息	样本 1	样本 2
第 1 层网络上的属性	0,tcp,http,SF,181,5 450,0,0,0	0,tcp,private,REJ,0,0,0,0,0
第 2 层网络上的属性	0,0,1,0,0,0,0,0,0,0,0,0,0	0,0,0,0,0,0,0,0,0,0,0,0,0
第 3 层网络上的属性	8,8,0.00,0.00,0.00,0.00,1.00,0.00,0.00,0.00	255,1,0.00,0.00,1.00,1.00,0.00,0.53,0.00
第 4 层网络上的属性	9,9,1.00,0.00,0.11,0.00,0.00,0.00,0.00,0.00,0.00	255,1,0.00,0.53,1.00,0.00,0.00,0.00,1.00,1.00
样本标签	正常	异常
MNDetector 检测结果	正常	异常
各基线方法检测结果	正常	正常

样本 1 和样本 2 在第 2 层上的属性高度相似, 而在第 3 层与第 4 层上的属性仅有少数字段存在差异, 仅在第 1 层上差异较为显著. 这说明样本 2 只在第 1 层上显著地体现出异常特性, 在其他属性字段的层面上与正常样本相似. 而各基线方法需汇总全部属性字段, 使得样本 2 中不体现异常特性的字段干扰了检测结果, 导致其被错误地识别为正常样本.

相应地, 对于 MNDetector, 尽管在第 2 层上未能成功将样本 2 识别为异常, 但由于样本 2 在其他层次上体现出了显著的异常特性, 因此得到的最终检测结果认定样本 2 为异常数据.

可以看出, 基于多层网络的异常检测本质上是将原始数据按照不同的属性层次进行划分, 弱化不同字段之间的关联, 消除异常属性中表现正常的部分对总体检测结果的影响, 从而解决属性干扰问题.

4.2.2 UNSW 的检测结果

从表 2 可以看出, 在 UNSW 数据集上, MNDetector 的召回率和精确率略低于 AnomalyDAE 和 AnomMAN, 但准确率和 $F1$ 分数更高, 因此在综合评价上 MNDetector 优于所有基线方法.

MNDetector 与基线方法的检测效果差异较小的原因可能是 UNSW 数据集中的异常特性在各属性字段上的分布较为均匀, 即异常数据在各属性字段层次上没有明显差异. 这使得 MNDetector 无法将异常特性不显著的属性字段排除在检测过程之外, 且并未成功分割出异常特性最集中的字段, 使得精确率与召回率偏低.

异常数据在 UNSW 数据集上的分布规律在 4.3 节所介绍的分层检测结果中也有体现.

4.3 分层检测

4.3.1 数据集分层结果

表 4 展示了所构建的多层网络中各层网络对应的属性字段数. 由于离散属性字段被转换为独热码, 因此各层字段数之和超过了原始属性字段数量.

Table 4 Number of Attribute Fields per Layer in Multiplex Networks

表 4 多层网络各层属性字段数

数据集	第 1 层	第 2 层	第 3 层	第 4 层	第 5 层	第 6 层
KDD	22	11	9	10		
NSL	22	11	9	10		
UNSW	131	31	5	5	18	
TOR	7	4	4	4	4	1

4.3.2 分层检测结果

表 5 展示了 MNDetector 在各数据集、各层次上的检测结果.

除了在 UNSW 数据集上的检测外, MNDetector 在其他数据集上的综合效果均为最优. 这说明相比于仅针对单一层次进行检测, 将全部层次进行综合带来了更多的信息, 有助于获得更优的检测结果. 此外, 虽然模型各层检测效果不及总体效果, 但在部分层次上依旧相对准确, 这说明异常性质显著的属性字段已经提供了足够多的信息, 能够支撑相对准确的检测.

与表 2 的检测结果相似, 在 UNSW 数据集上, 模型总体检测效果并不完全优于各层检测效果. 这验证了其异常特性在各属性字段上表现较为平均, 不在某一层次上显著突出. 在进行细粒度划分时, 模型在各层上的表现相差不大, 导致综合检测并不完全优于分层检测.

此外, 可以看出 KDD 和 NSL 数据集对应的第 2 层, 以及 TOR 数据集对应的第 3~5 层上的检测结果极差, 而 UNSW 数据集上却没有这种现象. 为对此进行进一步分析, 需考察所构建的多层网络的具体结构. 表 6 展示了各层网络包含的边数目.

结合分层语义以及表 6 可以看出, 上述层次相关的属性字段包含是否登录成功、是否为超级用户等信息. 这些字段与对于单一的访问行为而言属于基础性信息, 无法体现访问目的, 不具有异常区分度. 因此, 在这些层次上边的数目远大于其他层次, 形成了近乎全连接的图. 这意味着全部访问在相关属性字段上的值几乎一致, 在这些层次上难以学习到异常属性分布, 导致检测精确率数值为 1 但综合效果

Table 5 Layer-Wise Detection Results on Various Datasets

表 5 各数据集上的分层检测结果

网络层次	KDD				NSL			
	准确率	召回率	精确率	*F*1 分数	准确率	召回率	精确率	*F*1 分数
第 1 层	0.995 9	0.948 1	0.969 7	0.958 8	0.995 8	0.941 5	0.973 9	0.957 4
第 2 层	0.950 4	0.007 4	**1.00 0**	0.014 7	0.950 7	0.014 8	**1.00 0**	0.029 2
第 3 层	0.995 6	0.964 4	0.949 0	0.956 6	0.995 1	0.963 7	0.940 7	0.952 1
第 4 层	0.996 5	0.954 1	0.976 3	0.965 1	0.996 7	0.956 3	0.976 6	0.966 3
综合检测	**0.998 0**	**0.979 3**	0.980 2	**0.979 7**	**0.998 1**	**0.981 7**	0.980 8	**0.981 2**
网络层次	TOR				UNSW			
	准确率	召回率	精确率	*F*1 分数	准确率	召回率	精确率	*F*1 分数
第 1 层	0.995 2	0.948 1	0.955 2	0.951 6	0.996 1	0.941 7	**0.979 2**	0.960 1
第 2 层	0.996 3	0.958 5	0.966 4	0.962 4	0.992 5	0.883 3	0.963 6	0.921 7
第 3 层	0.962 4	0.351 9	0.772 4	0.483 5	0.991 3	0.925 0	0.902 4	0.913 6
第 4 层	0.950 6	0.011 9	**1.00 0**	0.023 5	0.994 3	0.939 2	0.945 5	0.942 3
第 5 层	0.951 9	0.045 2	0.859 2	0.085 9	0.993 2	0.933 3	0.930 2	0.931 7
第 6 层	0.993 7	0.966 7	0.912 6	0.938 9				
综合检测	**0.997 5**	**0.974 8**	0.975 5	**0.975 1**	**0.996 8**	**0.962 5**	0.973 9	**0.968 2**

注：黑体数值表示对应指标下的最优检测结果.

Table 6 Number of Edges in Each Layer of Network

表 6 各层次网络中边的数目

数据集	第 1 层	第 2 层	第 3 层	第 4 层	第 5 层	第 6 层
KDD	6 855	**455 326**	15 076	46 097		
NSL	6 770	**466 692**	15 765	40 997		
UNSW	15 154	9 893	15 225	17 965	10 296	
TOR	14 231	76 892	**263 183**	**468 028**	**463 203**	52 700

注：黑体数值表示对应层次下显著偏大的边数目.

Table 7 *F*1 Scores of Ablation Experiments

表 7 消融实验的 *F*1 分数

实验设置	数据集	
	KDD	TOR
不采用重构损失	0.950 1	0.949 5
不采用判别结果	0.951 3	0.940 2
不采用同层路径	0.878 4	0.796 3
不采用跨层路径	0.928 8	0.948 0
完整的 MNDetector	**0.982 1**	**0.973 9**

注：黑体数值表示对应数据集下的最优检测结果.

极差. 而在综合检测中, 此类层次的结果被屏蔽, 从而杜绝了不体现异常特性的属性字段对检测造成的干扰.

表 6 同样展示出, UNSW 数据集所构建的多层网络中不存在近似全连接的层次, 即多层网络结构并未将干扰检测的属性字段排除, 这也是其综合检测结果相较于基线方法提升不高的原因.

4.4 消融实验

以 *F*1 分数作为性能评价指标, 选取 KDD 和 TOR 数据集分别作为入侵检测和匿名流量检测的代表进行消融实验. 表 7 展示了消融实验的结果.

由表 7 不难看出, 从节点嵌入的角度而言, 综合同层与跨层路径能够有效提升检测效果; 从结果融合的角度而言, 综合利用生成器和判别器的检测效果优于单独使用的效果. 不同数据集上仅采用生成器或判别器的检测效果互有优劣, 这可能是由于异常样本同时在结构信息和属性信息上体现出异常, 而这二者分别是生成器和判别器的侧重点.

当仅采用层内路径进行检测时, 多层网络结构退化为失去跨层关联的分层网络, 引起了检测效果的降低. 而当仅采用跨层路径时, 模型对同一层次内节点关联的挖掘严重不足, 引起了最大程度的效果降低.

5 结 论

当前频繁出现的网络安全威胁的核心问题是异常访问. 访问数据的属性包含反映不同方面特性的字段, 然而只有部分字段体现出显著的异常特性. 现有检测方法通常采用全部属性字段进行统一检测,

引入了区分度较弱的字段,造成属性干扰问题,影响检测效果. 为解决这一问题,将多层网络结构引入了异常访问检测领域,提出基于多层网络的异常访问检测方法 MNDetector. 利用相关性紧密的属性字段构建单层网络,在此基础上添加层间关联,形成多层网络结构. 随后基于跨层游走获取同层和跨层节点序列,实现节点表示. 最后在不同层次上分别训练 GAN,结合重构损失与判别结果共同实现异常检测.

实验验证了 MNDetector 在多个公开数据集上的检测效果超过当前的最优方法. 同时,分层检测结果验证了 MNDetector 能够有效融合各层检测结果,并能够排除不显著体现异常特性的属性字段的干扰. 此外,消融实验结果证实了 MNDetector 所采用的检测机制的有效性. 为深入探究其获得更佳检测效果的原因,本文还进行了案例研究,通过分析实验结果,详细说明了 MNDetector 的有效性和检测优势.

未来的研究计划从以下 3 个方面展开:

1)相比于单层结构,多层网络结构具有更高的复杂度,能够挖掘更多的深层关联. 拟引入互信息等度量方式,探索更复杂的多层网络异常检测方法.

2)多层网络结构能拆分数据集属性字段的语义层次. 拟对现有异常访问行为检测的公开数据集进行语义分析,结合具体场景增进对各属性字段的理解,并进一步对数据集进行优化与整合.

3)基于多层网络异常检测的有效性,拟结合异常访问行为的具体特征,分析异常特性显著或不显著的属性字段的语义含义,为进一步的检测提供指导.

作者贡献声明: 袁子淇提出论文总体思路并撰写论文; 孙庆赟对研究思路及论文进行调整与优化; 周号益对研究方法提出具体建议; 朱祖坤完成实验验证并撰写论文; 李建欣总体把控研究方向并提供指导性建议与工作支持.

参 考 文 献

[1] Zhang Yan, Li Jiatong, Song Xiaoyi, et al. Survey of IoT device security detection[J]. Journal of Computer Research and Development, 2023, 60(10): 2271−2290 (in Chinese)
(张妍, 黎家通, 宋小祎, 等. 物联网设备安全检测综述[J]. 计算机研究与发展, 2023, 60(10): 2271−2290)

[2] Liu Qixu, Chen Yanhui, Ni Jieshuo, et al. Survey on machine learning-based anomaly detection for industrial Internet[J]. Journal of Computer Research and Development, 2022, 59(5): 994−1014 (in Chinese)
(刘奇旭, 陈艳辉, 尼杰硕, 等. 基于机器学习的工业互联网入侵检测综述[J]. 计算机研究与发展, 2022, 59(5): 994−1014)

[3] Zhao Xiaoqing, Jiang Zhongyuan, Ma Jianfeng. A survey of deep anomaly detection for system logs[C/OL]//Proc of the Int Joint Conf on Neural Networks. Piscataway, NJ: IEEE, 2022 [2024-02-01]. https://ieeexplore.ieee.org/xpl/conhome/9891857/proceeding

[4] Liu Fucheng, Wen Yu, Zhang Dongxue, et al. Log2vec: A heterogeneous graph embedding based approach for detecting cyber threats within enterprise[C]//Proc of the 26th ACM Conf on Computer and Communications Security. New York: ACM, 2019: 1777−1794

[5] Guo Hongcheng, Guo Yuhui, Yang Jian, et al. LogLG: Weakly supervised log anomaly detection via log-event graph construction[C]//Proc of the 28th Int Conf on Database Systems for Advanced Applications. Berlin: Springer, 2023: 490−501

[6] Chang Wenjing, Yu Jianjun, Zhou Xiaojun. GADAL: An active learning framework for graph anomaly detection[C]//Proc of the 6th APWeb-WAIM Int Joint Conf on Web and Big Data. Berlin: Springer, 2023: 435−442

[7] Lin Haiping, Wu Chengwen, Masdari M. A comprehensive survey of network traffic anomalies and DDoS attacks detection schemes using fuzzy techniques[J]. Computers and Electrical Engineering, 2022, 104: 108466

[8] Hallo M, Suntaxi G. A survey on SQL injection attacks, detection and prevention techniques — A tertiary study[J]. International Journal of Security and Networks, 2022, 17(3): 193−202

[9] Chu Xiaokai, Fan Xinxin, Yao Di, et al. Cross-network embedding for multi-network alignment[C]//Proc of the Int World Wide Web Conf. New York: ACM, 2019: 273−284

[10] Zhang Hongming, Qiu Liwei, Yi Lingling, et al. Scalable multiplex network embedding[C/OL]//Proc of the 27th Int Joint Conf on Artificial Intelligence. 2018: 3082−3088. [2024-02-01]. https://www.ijcai.org/proceedings/2018/428

[11] Park C, Kim D, Han Jiawei, et al. Unsupervised attributed multiplex network embedding[C]//Proc of the 34th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2020: 5371−5378

[12] Liu Weiyi, Chen P, Yeung S, et al. Principled multilayer network embedding[C]//Proc of the IEEE Int Conf on Data Mining Workshops. Piscataway, NJ: IEEE, 2017: 134−141

[13] Grover A, Leskovec J. node2vec: Scalable feature learning for networks[C]//Proc of the 22nd ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2016: 855−864

[14] Chattopadhyay S, Ganguly D. Node2vec with weak supervision on community structures[J]. Pattern Recognition Letters, 2021, 150: 147−154

[15] Behrouz A, Seltzer M. Anomaly detection in multiplex dynamic networks: From blockchain security to brain disease prediction[J]. arXiv preprint, arXiv: 2211.08378, 2022

[16] Chen Linghao, Li He, Zhang Wanyuan, et al. AnomMAN: Detect anomalies on multi-view attributed networks[J]. Information Sciences, 2023, 628: 1−21

[17] Shi J, Moura J. From DSP to GSP: Sampling in both domains[C]//Proc of the 56th Asilomar Conf on Signals, Systems, and Computers. Piscataway, NJ: IEEE, 2022: 240−245

[18] Perozzi B, Al-Rfou R, Skiena S. DeepWalk: Online learning of social representations[C]//Proc of the 20th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2014: 701−710

[19] Chanpuriya S, Musco C, Sotiropoulos K, et al. Deepwalking backwards: From embeddings back to graphs[C]//Proc of the 38th Int Conf on Machine Learning. New York: PMLR, 2021: 1473−1483

[20] Zhai Xingyu, Zheng Wenguang, Xiao Yingyuan, et al. Point-of-interest recommendation system based on deepwalk and tensor decomposition[C]//Proc of the 25th Int Conf on Computer Supported Cooperative Work in Design. Piscataway, NJ: IEEE, 2022: 867−872

[21] Wang Daixin, Cui Peng, Zhu Wenwu. Structural deep network embedding[C]//Proc of the 22nd ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2016: 1225−1234

[22] Guo Luxiang, You Zhuhong, Wang Lei, et al. A novel circRNA-miRNA association prediction model based on structural deep neural network embedding[J]. Briefings in Bioinformatics, 2022, 23(5): bbac391

[23] Kipf T, Welling M. Semi-supervised classification with graph convolutional networks[C/OL]//Proc of the 5th Int Conf on Learning Representations. 2017 [2024-02-01]. https://openreview.net/forum?id=SJU4ayYgl

[24] Masoumian A, Rashwan H, Abdulwahab S, et al. GCNDepth: Self-supervised monocular depth estimation based on graph convolutional network[J]. Neurocomputing, 2023, 517: 81−92

[25] Sun Qingyun, Li Jianxin, Yuan Haonan, et al. Position-aware structure learning for graph topology-imbalance by relieving under-reaching and over-squashing[C]//Proc of the 31st ACM Int Conf on Information and Knowledge Management. New York: ACM, 2022: 1848−1857

[26] Sun Qingyun, Li Jianxin, Peng Hao, et al. SUGAR: Subgraph neural network with reinforcement pooling and self-supervised mutual information mechanism[C]//Proc of the Web Conf. New York: ACM, 2021: 2081−2091

[27] Li Yuening, Huang Xiao, Li Jundong, et al. SpecAE: Spectral autoencoder for anomaly detection in attributed networks[C]//Proc of the 28th ACM Int Conf on Information and Knowledge Management. New York: ACM, 2019: 2233−2236

[28] Hu Jianke, Zhang Yin. NGAT: Attention in breadth and depth exploration for semi-supervised graph representation learning[J]. Frontiers of Information Technology & Electronic Engineering, 2022, 23(3): 409−421

[29] Cheng Ziqiang, Yang Yang, Jiang Shuo, et al. Time2graph+: Bridging time series and graph representation learning via multiple attentions[J]. IEEE Transactions on Knowledge and Data Engineering, 2023, 35(2): 2078−2090

[30] Sun Qingyun, Li Jianxin, Peng Hao, et al. Graph structure learning with variational information bottleneck[C]//Proc of the 36th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2022: 4165−4174

[31] Zhang Juntao, Lin Nanzhou, Zhang Xuelong, et al. Learning concept prerequisite relations from educational data via multi-head attention variational graph auto-encoders[C]//Proc of the 15th ACM Int Conf on Web Search and Data Mining. New York: ACM, 2022: 1377−1385

[32] Wang Hongwei, Wang Jia, Wang Jialin, et al. GraphGAN: Graph representation learning with generative adversarial nets[C]//Proc of the 32nd AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2018: 2508−2515

[33] Chen Bofeng, Li Jingdong, Lu Xingjian, et al. Survey of deep learning based graph anomaly detection methods[J]. Journal of Computer Research and Development, 2021, 58(7): 1436−1455 (in Chinese)
(陈波冯, 李靖东, 卢兴见, 等. 基于深度学习的图异常检测技术综述[J]. 计算机研究与发展, 2021, 58(7): 1436−1455)

[34] Song Junxian, Wen Rong, Yan Wenjing. Identification of traffic accident clusters using Kulldorff's space-time scan statistics[C]//Proc of the IEEE Int Conf on Big Data. Piscataway, NJ: IEEE, 2018: 3162−3167

[35] Neil J, Hash C, Brugh A, et al. Scan statistics for the online detection of locally anomalous subgraphs[J]. Technometrics, 2013, 55(4): 403−414

[36] Wang Chunpai, Neill D, Chen Feng. Calibrated nonparametric scan statistics for anomalous pattern detection in graphs[C]//Proc of the 36th AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2022: 4201−4209

[37] Ma Xiaoxiao, Wu Jia, Xue Shan, et al. A comprehensive survey on graph anomaly detection with deep learning[J]. IEEE Transactions on Knowledge and Data Engineering, 2023, 35(12): 12012−12038

[38] Yu Wenchao, Cheng Wei, Aggarwal C, et al. NetWalk: A flexible deep embedding approach for anomaly detection in dynamic networks[C]//Proc of the 24th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2018: 2672−2681

[39] Yang Linxiao, Cheung N, Li Jiaying, et al. Deep clustering by Gaussian mixture variational autoencoders with graph embedding[C]//Proc of the IEEE/CVF Int Conf on Computer Vision. Piscataway, NJ: IEEE, 2019: 6439−6448

[40] Ding Kaize, Li Jundong, Agarwal N, et al. Inductive anomaly detection on attributed networks[C/OL]//Proc of the 29th Int Joint Conf on Artificial Intelligence. 2020: 1288−1294. [2024-02-01]. https://www.ijcai.org/proceedings/2020/179

[41] Fan Ganghui, Ma Yong, Mei Xiaoguang, et al. Hyperspectral anomaly detection with robust graph autoencoders[J/OL]. IEEE Transactions on Geoscience and Remote Sensing, 2022, 60. [2024-02-01]. https://ieeexplore.ieee.org/document/9494034

[42] Fan Haoyi, Zhang Fengbin, Li Zuoyong. AnomalyDAE: Dual autoencoder for anomaly detection on attributed networks[C]//Proc of the 45th Int Conf on Acoustics, Speech, and Signal Processing. Piscataway, NJ: IEEE, 2020: 5685−5689

[43] Moustafa N, Slay J. UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)[C/OL]//Proc of the Military Communications and Information Systems Conf. Piscataway, NJ: IEEE, 2015 [2024-02-01]. https://ieeexplore.ieee.org/abstract/document/7348942

[44] Sakai H, Nakata M. Rough set-based rule generation and apriori-based rule generation from table data sets: A survey and a combination[J].

CAAI Transactions on Intelligence Technology, 2019, 4(4): 203−213

[45] Tavallaee M, Bagheri E, Lu Wei, et al. A detailed analysis of the KDD CUP 99 data set[C/OL]//Proc of the IEEE Symp on Computational Intelligence for Security and Defense Applications. Piscataway, NJ: IEEE, 2009 [2024-02-01]. https://ieeexplore.ieee.org/document/5356528

[46] Lashkari A, Gil G, Mamun M, et al. Characterization of Tor traffic using time based features[C/OL]//Proc of the 3rd Int Conf on Information Systems Security and Privacy. 2017 [2024-02-01]. https://www.scitepress.org/Link.aspx?doi=10.5220/0006105602530262

[47] Wang Hongbo, Zhao Guangzhou, Gu Hong. A fast training method for OC-SVM based on the random sampling lemma[C]//Proc of the 6th Int Conf on Natural Computation. Piscataway, NJ: IEEE, 2010: 824−827

[48] Zong Bo, Song Qi, Min M, et al. Deep autoencoding Gaussian mixture model for unsupervised anomaly detection[C/OL]//Proc of the 6th Int Conf on Learning Representations. 2018 [2024-02-01]. https://openreview.net/forum?id=BJJLHbb0-

[49] Schlegl T, Seeböck P, Waldstein S, et al. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery[C]//Proc of the 25th Int Conf on Information Processing in Medical Imaging. Berlin: Springer, 2017: 146−157

[50] Donahue J, Krähenb P, Darrell T. Adversarial feature learning[C]//Proc of the 5th Int Conf on Learning Representations. 2017 [2024-02-01]. https://openreview.net/forum?id=BJtNZAFgg



Yuan Ziqi, born in 1997. PhD candidate. Student member of CCF. His main research interests include graph anomaly detection, network big data analysis, and data mining.

袁子淇, 1997 年生. 博士研究生. CCF 学生会员. 主要研究方向为图异常检测、网络大数据分析、数据挖掘.



Sun Qingyun, born in 1996. PhD, assistant professor. Member of CCF. Her main research interests include graph machine learning, network big data analysis, and anomaly detection.

孙庆赟, 1996 年生. 博士, 助理教授. CCF 会员. 主要研究方向为图机器学习、网络大数据分析、异常检测.



Zhou Haoyi, born in 1991. PhD, assistant professor. Member of CCF. His main research interests include time-series forecasting, task decision-making, and AI safety.

周号益, 1991 年生. 博士, 助理教授. CCF 会员. 主要研究方向为时间序列预测、任务决策、人工智能安全.



Zhu Zukun, born in 2001. Master candidate. His main research interests include anomaly detection and federated learning.

朱祖坤, 2001 年生. 硕士研究生. 主要研究方向为异常检测、联邦学习.



Li Jianxin, born in 1979. PhD, professor. Distinguished member of CCF. His main research interests include big data analysis and processing, network big data computing, and scientific computing.

李建欣, 1979 年生. 博士, 教授.CCF 杰出会员. 主要研究方向为大数据分析与处理、网络大数据计算、科学计算.