Modern Engineering and Application 现代工程与应用, 2026, 4(3), 25-41
Published Online August 2026.
https://ufascience.com/mea/index

一种基于知识库的智能安全服务优化方法

周欣然

深圳大学经济学院，深圳 518060，广东，中国

摘要：网络安全知识库对网络安全数据进行了标准化与整合，为实时网络安全防护方案提供了可靠的基础。然而，当前对网络安全知识库的研究主要集中在其构建上，而利用知识库优化面向实时网络安全防护的智能安全服务这一潜力仍有待深入挖掘。因此，如何有效利用网络安全领域海量的历史知识并建立实时更新的反馈机制，从而提升安全服务对恶意流量的检测能力，已成为一个重要课题。本文的贡献有四个方面：第一，设计了一个反馈接口，用攻击流量特征、网络服务功能（NSF）的检测结果以及系统资源占用等信息更新知识库；第二，提出一种把 PageRank 与 RandomForest 相结合的特征选择方法，用以识别知识库中影响力较大的特征并动态融入各 NSF；第三，提出一种把图注意力网络（GAT）与深度强化学习（DRL）相结合的路径选择方法，用以学习知识库的局部知识并确定服务功能链（SFC）内的最优流量路径；第四，实验结果表明，知识库能够依据反馈信息实时更新，优化后的服务在准确率、召回率与 F1 分数上均超过 96%。与预设路径以及采用深度 Q 网络（DQN）方法选出的路径相比，本文方法把恶意流量检测率平均分别提升 12.4%与 4.6%，把路径的恶意流量总检测能力（TMTDC）分别提升 18.1%与 11.5%，并显著降低了路径检测时延。经验证，本文提出的智能安全优化方法能够实时监测恶意流量、更新知识，并增强系统对恶意流量的检测能力。

关键词：网络安全知识库；特征选择；路径选择；知识反馈

An Intelligent Security Service Optimization Method Based on Knowledge Base

XinRan Zhou

School of Economics, Shenzhen University, Shenzhen 518060, Guangdong, China

Abstract: The network security knowledge base standardizes and integrates network security data, providing a reliable foundation for real-time network security protection solutions. However, current research on network security knowledge bases mainly focuses on their construction, while the potential to optimize intelligent security services for real-time network security protection requires further exploration. Therefore, how to effectively utilize the vast amount of historical knowledge in the field of network security and establish a feedback mechanism to update it in real time, thereby enhancing the detection capability of security services against malicious traffic, has become an important issue. Our contribution is fourfold. First, we design a feedback interface to update the knowledge base with information such as features of attack traffic, detection outcomes from network service functions (NSF), and system resource utilization. Second, we introduce a feature selection method that combines PageRank and RandomForest to identify influential

文章引用: 周欣然. 一种基于知识库的智能安全服务优化方法[J]. 现代工程与应用, 2026, 4(3): 25-41.
DOI: https://doi.org/10.61784/mea2029

features in the knowledge base and dynamically incorporate them into the NSFs. Third, we propose a path selection method that combines graph attention network (GAT) and deep reinforcement learning (DRL) to learn the local knowledge of the knowledge base and determine the optimal traffic path within the Service Function Chains (SFC). Finally, experimental results demonstrate that the knowledge base can be updated in real time according to feedback information, and the optimized service achieves an accuracy, recall, and F1 score exceeding 96%. Compared to preset paths and paths selected using the deep Q-network (DQN) method, our proposed method increases the malicious traffic detection rate by an average of 12.4% and 4.6%, respectively, enhances the total malicious traffic detection capability (TMTDC) of the path by 18.1% and 11.5%, and significantly reduces path detection delay. It has been verified that the proposed intelligent security optimization method can monitor malicious traffic in real time, update knowledge, and enhance the system's detection capability against malicious traffic.

Keywords: Network security knowledge base; Feature selection; Path selection; Knowledge feedback

一、引言

随着互联网规模的迅速扩张以及物联网、云计算等新技术的出现，网络安全的边界日益模糊。传统网络安全系统难以有效应对源源不断的网络威胁。为此，研究者越来越多地把人工智能算法与安全服务相结合 [1]，力求实现异常流量的自动化检测。这类智能网络服务功能（NSF）通常以服务功能链（SFC）的形式交付给用户 [2]。然而，攻击者不断更新攻击策略，使流量特征更接近正常流量以逃避检测；而现有的多数智能安全服务方法较为僵化，缺乏对网络变化的动态适应能力 [3]。

网络安全知识库的出现使网络安全数据得以标准化与整合，能够在高度互联的终端之间持续存储并更新恶意流量信息，为构建实时网络安全防护方案奠定了可靠基础 [4]。网络安全知识库以知识图谱的形式存储信息，把实体之间的关系以图的方式刻画出来。该方法充当语义网络，具备高效的查询与展示能力以及灵活的存储与更新能力。

然而，当前对网络安全知识库的研究主要集中在其构建上，而利用知识库优化面向实时网络安全防护的智能安全服务这一潜力仍有待深入挖掘。为此，本文提出一种基于网络安全知识库的智能安全服务优化方法，解决以下三项挑战。

第一项挑战是如何实时更新知识库中的恶意流量信息。为此，本文设计了一个反馈接口，把包括攻击流量特征、各 NSF 检测结果与系统资源占用在内的恶意流量信息回传至知识库以持续更新。

第二项挑战是如何从知识库中选取用于检测的恶意流量特征。本文采用 PageRank 与 RandomForest 相结合的特征选择方法：PageRank 用于在知识图谱中进行有效的知识推理与特征选择，RandomForest 则用于从特征数据集中筛选信息。

第三项挑战是如何利用知识库中的信息选择流量通过 SFC 的路径。本文用图注意力网络（GAT）优化深度强化学习（DRL）以选出最优检测路径。GAT 借助注意力机制，使模型能够更好地捕捉数据之间的关系信息与特征信息，从而实现对数据的深度分析与挖掘；DRL 则使智能体能够在无需人工启发式规则的情况下探索环境并从经验中学习。

本文的主要贡献如下：（1）设计了一个反馈接口，用攻击流量特征、各 NSF 检测结果与系统资源占用等信息更新知识库，该接口同时动态调整流量通过 SFC 的路径；（2）提出一种 PageRank 与 RandomForest 相结合的特征选择方法，识别知识库中影响力较大的特征并动态融入各 NSF，以提升其检测能力；（3）提出一种 GAT 与 DRL 相结合的路径选择方法，使系统能够学习知识库的局部知识并确定 SFC 内的最优流量路径；（4）在实验环境中开展离线训练与在线检测，系统比较了不同方案的性能，结果表明本文提出的基于网络安全知识

库的智能安全服务优化方法能够显著提升检测性能并降低路径检测时延。

本文其余部分安排如下：第二部分介绍相关工作；第三部分阐述智能安全服务优化方法；第四部分给出实验环境与结果；第五部分总结全文。

二、相关工作

网络安全知识库汇集了网络中各类恶意流量信息，可作为安全事件分析的重要数据来源，并为网络系统提供可靠的防御方案。国内外许多研究都聚焦于网络安全知识库与智能安全服务。

知识库可以知识图谱的形式展示数据。作为安全领域的专用知识图谱，网络安全知识图谱由节点与边构成大规模的安全语义网络，为现实安全世界中的各类攻防场景提供了直观的建模方法。为全面刻画分布式拒绝服务（DDoS）攻击，构建了 DDoS 攻击的恶意行为知识库，它由恶意流量检测知识库与网络安全知识库两部分组成：前者负责检测并分类 DDoS 攻击所产生的恶意流量，后者负责数据结构处理、恶意行为知识图谱创建与行为推理。建立的知识库主要包括知识构建与知识展示两部分：知识构建负责从结构化数据源与非结构化文本中抽取知识，知识展示则主要负责网络安全知识的可视化功能。提出了一种基于五元组模型的恶意行为知识库构建模型：首先用机器学习方法抽取并构建实体以获取网络安全知识，并用命名实体识别（NER）方法完成网络安全领域的知识抽取。

在智能安全服务领域，无线信道上的数据传输需要必要的安全措施，以防止攻击者窃听或篡改信息。为此，提出了一种密钥管理协议，用三个辅助密钥配合一个主密钥在网络内的三个不同层级上加密信息。此外，DDoS 攻击的检测与防御一直是网络安全领域的长期难题。

特征选择是提升 DDoS 攻击检测性能的关键环节。已有攻击检测工作在特征选择方面有以下代表性成果。用 Boruta 特征选择方法得到了一组典型的僵尸网络特征集，但单一特征选择算法无法充分挖掘数据特征与预测结果之间的关联，可能导致特征表达能力不足并影响在线检测性能。组合了多种特征选择算法：被三种及以上算法选中的特征为重要特征，仅被两种算法选中的特征为次重要特征，其余特征构成非重要特征集。用皮尔逊矩相关分析特征与类别标签之间的相关性，并用曲线下面积（AUC）度量各特征的重要性，仅使用前五个特征即取得了良好效果。提出了一种基于 RandomForest 的方法 RFUTE，用以处理偏标记学习（PLL）中的特征选择问题：先消除候选标签的歧义，再通过计算并排序森林中所有树的所有特征的信息熵总变化量来完成特征选择。PageRank 是由谷歌创建并应用的网页排序算法，主要用于评估网络中各节点的重要性，在好友推荐、网页搜索与链接分析等方面被广泛使用，但尚未被应用于特征选择领域 [5]。

SFC 路径选择旨在为用户提供高速、低时延的多样化网络功能定制服务，是提升检测性能的另一关键环节。提出了一种基于图神经网络构建 SFC 的算法，利用图神经网络中节点的表示，在其邻居节点的影响下构建灵活高效的 SFC。提出了一种面向物联网 DRL 的 SFC 动态编排框架，用以解决边缘云中虚拟网络功能（VNF）部署与端到端时延服务路径的优化问题。提出了一种面向安全 SFC 的深度 Q 网络（DQN）路径选择方法，但其恶意流量检测准确率有待提高、检测时延有待降低。把 DQN 与卷积神经网络（CNN）相结合，提出了一种 SFC 路径选择算法，该算法采用 CNN 近似状态—动作对的 Q 函数，可用卷积层与池化层提取数据特征并实现从数据结构中自动学习。针对多基站上部署网络切片的密集蜂窝网络场景，把 GAT 与 DRL 相结合，设计了一种智能实时的片间资源管理策略，以应对频繁的基站切换并满足不同业务需求的波动 [6-7]。

相关文献表明，当前知识库与智能安全服务研究存在若干关键问题。其一，网络安全知识库的多数研究主要着眼于构建，而利用其优化智能安全服务以实现实时网络安全防护的潜力基本未被挖掘。其二，现有特征选

择方法仅依赖 RandomForest 等特征工程算法，无法充分刻画数据特征的重要性，导致恶意行为特征表达不足。此外，GAT 虽能捕捉数据之间的关系信息以支持深度分析与数据挖掘，DRL 亦能增强 SFC 路径选择的适应性，但目前尚无研究把二者结合用于 SFC 路径选择。

上述空白促成了本文方法的提出：本文充分利用知识库中丰富的数据资源来优化智能安全服务，把基于图的特征选择方法 PageRank 与传统特征工程技术相结合，以更有效地挖掘攻击特征；同时把 GAT 与 DRL 引入 SFC 路径选择以提升其效率，力求检测多种类型的恶意攻击、提高检测准确率并降低检测时延，具有较强的实用价值。

三、智能安全服务优化方法

本文提出一种智能安全服务框架，并按功能划分为四个模块。第（一）节概述整个框架；第（二）节详述数据反馈模块；第（三）节阐述知识库模块的组成；第（四）节说明 NSF 特征选择模块与 SFC 路径选择模块的实现。

（一）总体框架

本研究提出的智能安全服务框架如图 1 所示，包含四个关键模块：数据反馈模块、知识库模块、安全策略推理模块与服务功能模块。

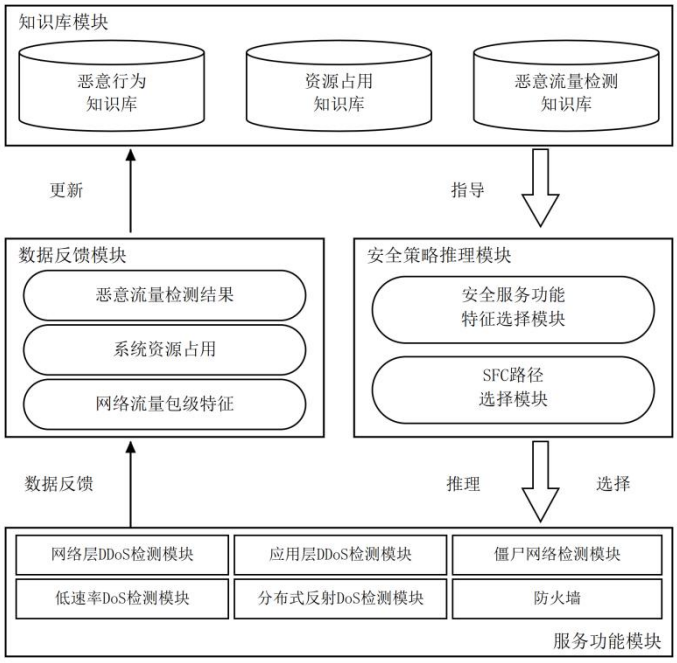

Figure 1 Intelligent Security Service Framework

图 1 智能安全服务框架

数据反馈模块的主要功能是周期性地从各 NSF 收集反馈数据、存入知识库并更新知识库，以指导安全策略的推理。所收集的数据包括进入 SFC 的网络流量包级特征、各 NSF 的恶意流量检测结果以及系统资源占用情况。

知识库模块由恶意行为知识库、恶意流量检测知识库与资源占用知识库构成。恶意行为知识库主要包括 DDoS 攻击的流级特征数据集与包级特征数据集，以及恶意行为特征图。恶意流量检测知识库存储服务功能模块的检测结果，如准确率、恶意流量检测率、检测能力与检测时间等。资源占用知识库跟踪服务功能模块的系

统资源占用情况，包括 CPU 占用率、内存占用率、磁盘占用率、丢包率等。

安全策略推理模块由NSF特征选择模块与SFC路径选择模块构成。本文利用知识库中存储的知识设计算法，动态调整流量需要经过的 NSF。

服务功能模块把各类恶意流量检测方法虚拟化为 NSF，包括网络层 DDoS（NetDDoS）检测模块、应用层 DDoS（AppDDoS）检测模块、僵尸网络（Botnet）检测模块、低速率 DoS（LDDoS）检测模块、分布式反射 DoS（DRDoS）检测模块与防火墙。各检测方法组合构成 SFC 的检测路径，不同的检测路径对安全性能有不同的影响 [8]。

（二）数据反馈模块

数据反馈模块负责收集进入SFC的网络流量包级特征、各NSF的恶意流量检测结果以及系统资源占用情况，并把这些信息实时反馈至知识库，从而实现对流量通过 SFC 路径的动态调整。其流程如图 2 所示。

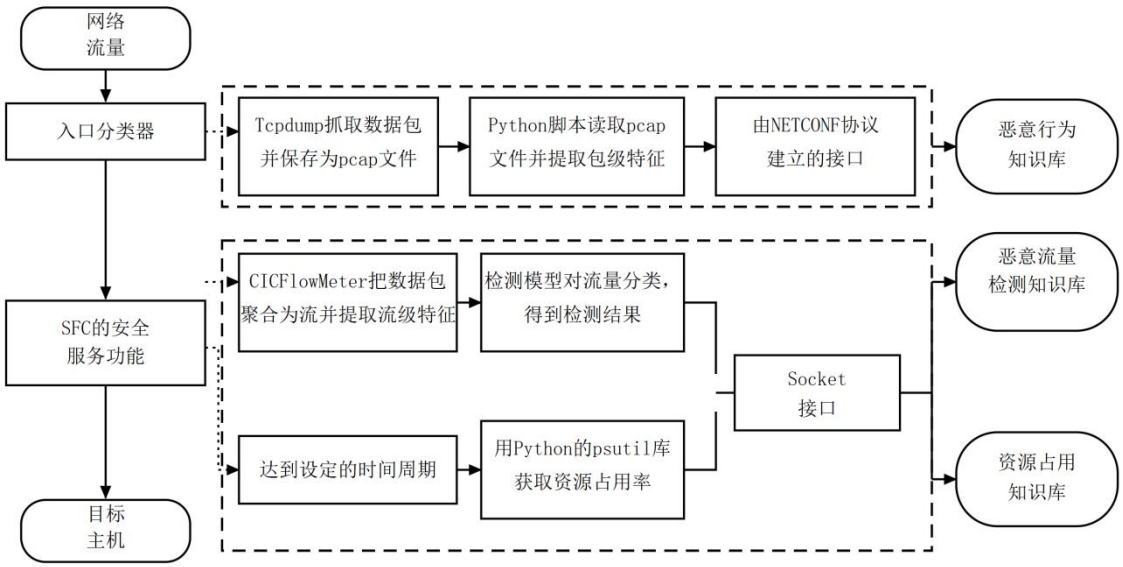

Figure 2 Data Feedback Diagram

图 2 数据反馈流程

本文在入口分类器处用 Tcpdump 抓包工具捕获进入 SFC 的 DDoS 攻击流量，保存为 pcap 文件，再用 Python 脚本解析以提取包级特征。这些特征保存为逗号分隔值（CSV）文件，并通过基于网络配置（NETCONF）协议建立的接口反馈至恶意行为知识库。在这一步骤中，恶意行为知识库充当客户端，而负责从攻击流量中提取包级特征的入口分类器充当服务端：恶意行为知识库先向服务端订阅，特征提取完成后服务端自动把包级特征发送给已订阅的客户端；随后客户端用 Python 的 xml.dom 库解析经 NETCONF 协议接收到的包级特征信息，并保存至实时更新的恶意行为知识库。

各 NSF[10,21-24]采用 CICFlowMeter 流特征提取工具把五个相同的数据包聚合为一条流，从而提取详细的流级特征信息，支撑模型训练与在线部署，实现细粒度的攻击分类。与此同时，本文用 Python 的 psutil 库按特定时间间隔监测承载各 NSF 的 Docker 容器内的 CPU、内存等资源占用情况。各 NSF 的分类检测结果与系统资源占用通过 Socket 接口分别回传至恶意流量检测知识库与资源占用知识库。这两个知识库充当服务端，采用多线程 Socket 接口接收各检测模块的反馈；各 NSF 作为客户端，通过 Socket 传输流量检测信息与资源占用情况以更新两个知识库。各模块的流量检测数据包括准确率、恶意流量检测率、检测能力与检测时间，资源占用指

标包括 CPU 占用率、内存占用率、磁盘占用率与丢包率。

知识库通过实时反馈机制持续更新：流量监测模块检测到的攻击特征与系统资源占用等数据被回传至系统。这一动态更新策略保证知识库始终保持最新，使系统能够对新出现的威胁与攻击模式作出及时响应。本文把恶意行为库中存储的 DDoS 攻击流量包级特征输入 SFC 路径选择模块，作为路径选择算法的输入；恶意流量检测知识库的检测结果与资源占用知识库的 CPU 占用率则作为路径选择算法的参数。该方法旨在依据实时更新的知识库数据动态调整服务路径，以确定由各 NSF 组合而成的最优检测路径。

（三）知识库模块

知识库模块由恶意行为知识库、恶意流量检测知识库与资源占用知识库构成，与数据反馈模块回传至知识库的三类信息一一对应。各知识库与其他模块的结构关系如图 3 所示。

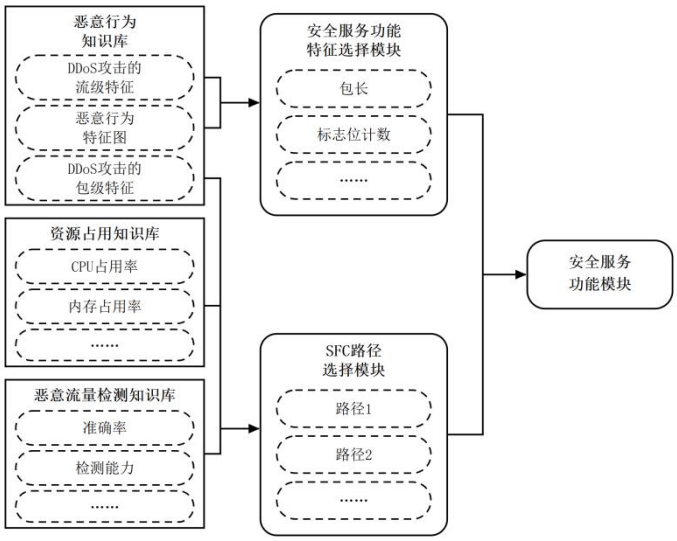

Figure 3 Relationship between the Knowledge Base and Other Modules

图 3 知识库与其他模块之间的关系

恶意行为知识库的主要组成部分包括 DDoS 攻击的流级特征数据集、恶意行为特征图与 DDoS 攻击的包级特征数据集。DDoS 攻击的流级特征数据集涵盖 CICFlowMeter 提取的 83 个特征属性，示例见表 1。

Table 1 Flow-level Features of DDoS Attacks

表 1 DDoS 攻击的流级特征

特征类型	特征名称
流标识特征	Protocol
标志位特征	FIN Flag Count, SYN Flag Count, RST Flag Count, ACK Flag Count, Fwd PSH Flag, Bwd URG Flag
流首部特征	Fwd Header Length, Bwd Header Length, FWD Init Win Bytes
时间特征	Fwd IAT Min, Fwd IAT Max, Fwd IAT Mean, Fwd IAT Std, Fwd IAT Total, Flow IAT Min, Flow IAT Mean, Flow IAT Std, Flow duration, Idle Min, Idle Max, Idle Mean
载荷特征	Total Fwd Packet, Total Bwd Packet, Total Length of Fwd Packet, Packet Length Min, Bwd Packet Length Max, AVG Bwd Segment Size, Flow Byte/s, Fwd Packets/s, Bwd Avg Bytes/Bulk

恶意行为特征图给出了 DDoS 攻击的详细分类与特征匹配。例如在 DRDoS 攻击类别下有 Chargen、NTP 与 TFTP 等子类，每个子类都具有区别于正常流量的独特特征（如 Idle Mean）。DDoS 攻击的流级特征数据集与恶意行为特征图作为 NSF 特征选择模块的输入，为其提供数据支撑。

表 2 列出了用于识别 DDoS 攻击的包级特征，主要以基于熵的特征为主，包括 IP 与端口的熵、包速率、数据包的条件熵等。DDoS 攻击通常比正常流量产生更高的流量速率，因此本文选取包速率与字节速率作为特征。DDoS 攻击发生时，大量数据包会指向同一目的 IP 地址，因此通过测量指定时间窗口内 IP 地址的熵即可有效识别 DDoS 攻击的发生。此外，计算端口号的熵以及端口与 IP 相结合的条件熵，可有效区分不同类型的 DDoS 攻击流量。这些特征作为 SFC 路径选择模块的输入，为动态调整服务路径以确定最优检测路径提供数据支撑。

Table 2 Packet-level Features of DDoS Attacks

表 2 DDoS 攻击的包级特征

特征	说明
包速率	每秒转发的平均数据包数
字节速率	每秒转发的平均字节数
平均包长	数据包的平均长度
源 IP 熵	源 IP 地址的熵
目的 IP 熵	目的 IP 地址的熵
TTL 熵	数据包生存时间 TTL 的熵
TCP 源端口熵	TCP 数据包源端口的熵
TCP 目的端口熵	TCP 数据包目的端口的熵
UDP 源端口熵	UDP 数据包源端口的熵
UDP 目的端口熵	UDP 数据包目的端口的熵
包长熵	数据包长度的熵
H(Sip|Dip)	给定目的 IP 时源 IP 的熵
H(Sip|Dport)	给定目的端口时源 IP 的熵
H(Dport|Dip)	给定目的 IP 时源端口的熵
包数方差	单位时间间隔内数据包数量的方差

表 3 列出了资源占用知识库的内容，包括 CPU 占用率、内存占用率、磁盘占用率与丢包率等关键指标。通过分析 CPU 占用、内存占用与磁盘占用，可有效识别并处理系统性能下降或故障的成因；监测丢包率则有助于发现网络连通性问题，便于及时排查与解决。这些指标对于在资源占用保持在可接受范围内的前提下构建最优 SFC 路径至关重要，并被用作路径选择算法的参数。

Table 3 Resource Usage Knowledge Base

表 3 资源占用知识库

名称	说明	单位
cpu_usage	CPU 占用率	%
cpu_freq	CPU 频率	MHz

memory_total	系统内存总量	GB
memory_available	可用内存	GB
memory_usage	内存占用率	%
disk_usage	磁盘占用率	%
disk_left	磁盘剩余空间	GB
packet_loss_rate	丢包率	%

恶意流量检测知识库的核心内容是各 NSF 的检测结果，如表 4 所示。这些检测结果与流量特征相对应，反映各模块在给定时刻的准确率、检测率、检测能力与检测时间，体现出检测模块识别当前流量模式的有效性。这些结果被用作路径选择算法的参数，提供关键的数据支撑。

Table 4 Malicious Traffic Detection Knowledge Base

表 4 恶意流量检测知识库

内容	说明
模块名称	检测模块的名称
时间戳	检测发生的时间戳
准确率	被正确预测的流量占流量总数之比
检测率	被正确预测的攻击占全部被预测为攻击者之比
检测能力	攻击总数与未被检出攻击数之比
检测时间	检测模块所耗费的时间

恶意行为知识库、资源占用知识库与恶意流量检测知识库并非彼此孤立运行，而是以流量编号作为共同主键实现数据关联：同一周期内收到的反馈信息被汇聚到对应的流量编号之下。这一做法便于检索流量特征、识别是哪个 NSF 检测到该流量，并可获取各检测模块的检测结果与资源占用数据。实体关系如图 4 所示。

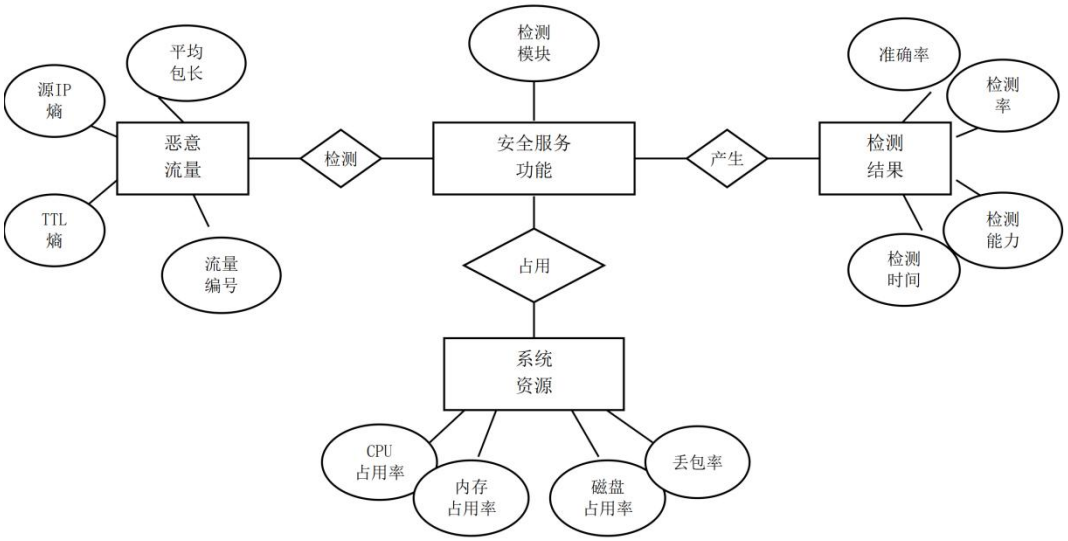

Figure 4 Entity Relationship Diagram

图 4 实体关系图

为保持知识库的时效性，本文实施了自动过期机制：每条数据条目都被赋予时间戳，超过预设阈值的条目将被标记为过期，或归档以备进一步分析，或从活跃使用中移除。此外，系统还会定期审查知识库内容，依据当前网络状况与威胁态势清除或更新过期信息。

（四）安全策略推理模块

1. NSF 特征选择模块

特征选择在机器学习中至关重要，尤其是通过从原始特征数据中筛选子集来提升检测模块的性能。本节提出一种把 PageRank 特征排序方法与 RandomForest 算法相结合的新型特征选择方法，该方法利用恶意行为知识库中的恶意行为特征图与 DDoS 攻击流级特征数据集，旨在评估图中各攻击所对应特征的重要性，找出对攻击分类影响显著的流级特征，并把这些信息融入各 NSF 以提升检测效率与检测能力。

DDoS 攻击的恶意行为特征图表明，每类攻击与特定类型的流量特征之间存在相关性。然而仅依赖 RandomForest 等特征工程算法无法充分挖掘数据特征的重要性，导致恶意行为特征表达不足。PageRank 是一种著名的基于图的排序算法，主要通过分析网络内的链接结构对网页排序。本研究把 PageRank 算法用于在恶意行为特征图上进行有效推理与特征选择。因此，本文把 PageRank 与 RandomForest 结合用于特征选择：PageRank 用于恶意行为特征图中的知识推理与特征选择，RandomForest 用于从 DDoS 攻击流级特征数据集中筛选信息。

然而，知识库中的恶意行为特征图缺少特征之间的关系，而这正是 PageRank 在强连通有向图上随机游走所必需的。为此，需要把该图从树形结构转换为强连通有向图，转换公式如下：

$$A：X \rightarrow Y，X \rightarrow Z \tag{1}$$

$$B：Y \Leftrightarrow Z \tag{2}$$

式中，X 表示攻击类型，Y 与 Z 表示流级特征。图由状态 A 转换为状态 B。例如对于 DRDoS 攻击。

RandomForest 特征选择算法评估每类 DDoS 攻击的重要性，并按显著程度对特征排序。该方法评估每个特征对随机森林中每棵树的贡献，计算其平均贡献，并比较各特征之间的重要性。

把 PageRank 与 RandomForest 结合以计算特征重要性得分的伪代码见算法 1。恶意行为特征图包含代表 DDoS 攻击特征的 83 个节点，因此节点数设为 n = 83。为提升 PageRank 的可靠性，参数设置如下：阻尼系数 d = 0.85，计算精度 σ = 1e−5，最大迭代次数 Q = 100，取值参照文献的建议。一阶马尔可夫链转移矩阵 M 由映射后的强连通特征图导出，x(1)表示特征重要性得分的初始分布矩阵，按各特征节点重要性相等初始化，即取 1/83。

PageRank 与 RandomForest 方法相结合后，各特征节点最终得分的计算公式如下：

$$score = \alpha RF + \beta PR \tag{3}$$

$$0 < \alpha, \beta, PR, RF < 1 \tag{4}$$

$$\alpha + \beta = 1 \tag{5}$$

式中，PR 与 RF 分别表示 PageRank 算法与 RandomForest 算法给出的特征重要性得分；α 与 β 分别表示 RF 与 PR 的参与系数，取值介于 0 至 1 之间。随后依据这些得分对特征重要性排序，并选出排名靠前的流级特征。用这一精炼后的特征集重新训练服务功能模块中的攻击检测模型，即可获得优化后的检测性能。

2. SFC 路径选择模块

SFC 路径选择模块训练一个名为 GAT+DRL 的路径选择模型，把 GAT 与 DRL 相结合以选出最优检测路径。本文首先把恶意行为知识库中 DDoS 攻击的包级特征数据集输入 GAT，由 GAT 学习数据内部的特征表示与结构

关系：随后把聚合后的特征传递给 DRL，由 DRL 把路径选择任务建模为马尔可夫决策过程。DRL 通过最大化奖励来选择最优检测路径，该奖励由恶意流量检测知识库的检测结果与资源占用知识库的 CPU 占用率共同构成。

GAT 是一种在数据之间引入注意力机制的图神经网络模型，能够增强其捕捉数据之间关系信息与特征信息的能力，从而具备深度分析与数据挖掘能力。它可以挖掘知识库中恶意流量的高层特征信息，DRL 则可基于这些高层特征更有效地选择路径。另一方面，DRL 把深度学习的感知能力与强化学习的决策能力结合起来，DQN 即为其中的典型代表：它用深度神经网络近似 Q 函数，网络以环境状态为输入，预测该状态下所有可能动作的 Q 值。

恶意行为知识库中大量来自不同 DDoS 攻击的包级特征数据首先被输入 GAT。在 GAT 中，每个攻击节点用聚合函数汇聚其邻居节点传来的特征信息，所汇聚的信息再经非线性更新函数进行变换；该迭代过程重复多次，从而为每个攻击节点生成结果特征。

输入节点的特征信息记为 $h = \{h1, h2, \cdots, hN\}$，$hi \in RF$，其中 N 表示输入恶意流量的总数，F 表示每条流量实例的特征维度。随后应用自注意力机制计算节点 i 与其邻居节点 j 之间的注意力得分。

$$ei,j = a(Whi, Whj) \tag{6}$$

式中，$W \in RF' \times F$ 表示可训练的共享线性变换矩阵。该矩阵作用于每个流量节点，把原始特征空间变换到更高层的特征空间，从而增强节点的表达能力；函数 a(•)用于计算两个节点（向量）之间的相关性。

节点 j 对节点 i 的重要性由 ei,j 确定。为使不同节点之间的系数具有可比性，本文用 softmax 函数归一化，从而得到标准化的注意力得分。

$$\alpha i,j = softmax(ei,j) = exp(LeakReLu(a[Whi \| Whj])) / \Sigma k \in Ni\ exp(LeakReLu(a[Whi \| Whk])) \tag{7}$$

式中，$\|$ 表示两个变换后节点的拼接；a 为可训练的参数向量，称为注意力参数向量，其规模为 $2 \times F'$（变换后嵌入维度的两倍），用于学习节点与其邻居之间的相对重要性；整个表达式 a(Whi, Whj)表示 a 与变换后节点之间的点积。

把每个邻居的特征向量乘以维度变换向量参数、按注意力得分加权，并经激活函数处理后，即可得到每个节点所对应的聚合特征。

$$h'i = \sigma(\Sigma j \in Ni\ \alpha i,jWhj) \tag{8}$$

基于 GAT 的聚合特征、恶意流量检测知识库中的检测结果以及资源占用知识库中的资源占用情况，本文构建了 DRL 环境，并把 SFC 路径选择问题建模为由状态空间、动作空间与奖励函数构成的马尔可夫决策过程（MDP）。

状态：在 DRL 框架中，状态表示智能体从环境中获得的信息。对于 SFC 路径选择问题，状态空间由 GAT 输出的聚合特征与恶意流量检测知识库中各检测模块的准确率构成，可表示为向量 $St = (h'i, \Phi)$，其中 h′i 表示上一阶段由 GAT 聚合得到的高层特征，Φ 表示各检测模块的检测准确率。

动作：在 DRL 中，NSF 的选择由流量特征与检测结果引导。因此动作空间 A 定义为可选 NSF 的集合 $A = \{f1, f2, \cdots, f6\}$，包括五个检测模块与防火墙，共六种方案。智能体执行一个动作时，即把对应的 NSF 加入路径列表；若动作为加入防火墙，则路径列表构建结束，本轮训练随之终止。

奖励：DRL 智能体通过接收来自外部环境的奖励信号来改进其表现。通常，当基于当前状态选中某个 NSF 时会给予正奖励，以强化该动作被选中的可能性，该正奖励与恶意流量检测知识库中的恶意流量检测能力以及资源占用知识库中的 CPU 占用率相关；反之，若所选动作重复了 SFC 路径中已包含的检测模块，则给予负奖励，以促使智能体探索其他决策。动作 at 的奖励函数定义如下：

$$rt = -\omega \text{（NSF 重复时）；} \quad rt = MTDC \cdot (1 - cpu\_usage) / n \text{（其他情形）} \tag{9}$$

式中，ω 为一个固定正数，MTDC 为所选 NSF 的恶意流量检测能力，cpu_usage 为 CPU 占用率，n 为数据包所经过的 NSF 数量。

在 DRL 模型的训练过程中，智能体在每次决策后采用 ε 一贪婪策略选择动作。ε 为 0 到 1 之间的数值：智能体以 ε 的概率选择已知的最优动作，以 $1-\varepsilon$ 的概率随机选择动作以探索未知情形。鉴于安全服务功能的多样性与复杂性，备选路径的数量可能非常庞大，此时人工设计备选路径并不现实，因而需要智能体探索环境并提升 SFC 路径选择的适应性，以便选出最优路径。执行动作后，相应的状态、动作、奖励与后继状态信息被存入经验池；系统定期从池中采样批量数据用于训练，并在训练过程中更新目标网络参数。训练中用于评估估计性能的损失函数定义如下：

$$J(\theta t) = E[(rt + \gamma Q(st, \text{argmax}_a Q(st+1, a|\theta t)|\theta' t) - Q(s, at|\theta t))^2] \tag{10}$$

式中，θt 与 $\theta' t$ 分别表示评估网络与目标网络的参数，γ 为折扣因子。损失函数定义为来自目标网络的目标 Q 值 $rt + \gamma Q(st, \text{argmax}_a Q(st+1, a|\theta t)|\theta' t)$与来自评估网络的估计 Q 值 $Q(s, at|\theta t)$之间的均方误差。最小化该损失函数可提升模型的评估性能。本文采用梯度下降算法更新评估网络的参数以优化损失函数，公式如下：

$$\theta + t = \theta t + \beta [rt + \gamma Q(st, \text{argmax}_a Q(st+1, a|\theta t)|\theta' t) - Q(s, at \ \theta t)] \nabla Q(s, at \ \theta t) \tag{11}$$

式中，$\theta + t$ 表示应用梯度下降算法后估计网络的更新参数，β 表示梯度下降更新过程中所用的步长参数。

算法 2 的伪代码给出了把 GAT 与 DRL 相结合以检测 DDoS 攻击的训练流程：先初始化环境与网络参数，再执行基于 GAT 的特征提取，让 DRL 智能体以 ε 一贪婪方式与环境交互，把经验存入回放记忆库，依据采样到的经验更新网络参数并定期更新目标网络，最后保存训练好的模型以用于恶意流量的在线检测。

四、实验结果

本部分对所提出的智能安全服务优化方法进行实验测试与结果分析。第（一）节介绍实验环境的搭建；第（二）节给出若干新的评价指标；第（三）节评估知识库数据的更新过程；第（四）节考察 NSF 特征选择算法与 SFC 路径选择算法的性能；第（五）节评估系统在不同路径下的在线检测性能；最后，为验证用实时反馈数据更新知识库对路径选择性能的影响，第（六）节基于检测时延与 TMTDC 评估知识库更新前后的效果。

（一）实验环境

本节用 VMware vSphere 搭建实验环境。各虚拟机运行 Ubuntu 18.04，分配 16 GB 内存与 30 GB 磁盘空间。实验装置包括一台知识库主机、两台分类器、四台转发器、一台用于产生攻击流量与正常流量的主机以及一台目标主机。软件实现方面，用 Open vSwitch 与 OpenDaylight 构建 SFC，并通过 Docker 实现各 NSF 的虚拟化部署。

本实验环境中部署了多种 NSF，包括 NetDDoS 检测模块、AppDDoS 检测模块、Botnet 检测模块、LDDoS 检测模块、DRDoS 检测模块与防火墙。为评估系统，本文用 Python 脚本与攻击工具对指定主机模拟 21 种攻击；正常流量数据则采集自 5G 环境下的语音通话、视频流与在线游戏等活动。

（二）评价指标

实验中采用五项评价指标衡量检测效果：混淆矩阵、准确率、精确率、召回率与 F1 分数。混淆矩阵展示模型预测与实际标签之间的关系。

此外，本文还定义了用于评估在线检测效果的新指标：恶意流量检测率、路径检测时延与恶意流量总检测能力（TMTDC）。恶意流量检测率量化检测后被正确识别的攻击流量比例。其中，i 为攻击流量类别，n 为攻

击流量类别总数，Ti 为类别 i 中被正确检出的实例数，Ai 为类别 i 中的实例总数。恶意流量检测率定义为：

$$\text{Detection Rate} = \Sigma^{n}_{i=1} Ti / \Sigma^{n}_{i=1} Ai \quad (12)$$

路径检测时延：对于一条 SFC 路径，路径检测时延反映其响应速度，是一项关键的性能评价指标。式（13）给出路径检测时延的计算公式：

$$\text{DelaySFC} = \Sigma Ci=1 \text{ dealy}(NSFc, NSFc+1) / C \quad (13)$$

式中，C 表示 SFC 的长度，$\Sigma Ci=1$ dealy(NSFc, NSFc+1)表示从第 c 个 NSF 到第(c + 1)个 NSF 的路径时延，即 SFC 路径上各段时延之和。

TMTDC：该指标衡量所选路径上的恶意流量检测能力，是评价路径质量的关键指标，定义为：

$$\text{TMTDC} = \Sigma Cc=0 \text{ MTDC}(NSFc) / C + \{\text{Average}[\text{MTDC}(NSFc)] \times \Sigma Cc=0 \text{ MTDC}(NSFc) / (\text{Average}[\text{MTDC}(NSFc)] - C)\} / C \quad (14)$$

式中，C 为路径列表中所选 NSF 的数量，MTDC(NSFc)表示第 c 个 NSF 的检测能力，$\Sigma Cc=0$ MTDC(NSFc)表示所选 NSF 检测能力之和，Average[MTDC(NSFc)]表示所选 NSF 的平均检测能力。检测能力指恶意流量总数与未被检出的恶意流量之比，定义为：

$$\text{MTDC} = 1 / (1 - \Sigma^{n}_{i=1} Ti / \Sigma^{n}_{i=1} Ai) \quad (15)$$

（三）知识库更新

服务功能模块中各检测模块的流量检测结果包括准确率、恶意流量检测率、检测能力与检测时间；资源占用指标包括 CPU 占用率、内存占用率、磁盘占用率与丢包率。这些指标通过 Socket 接口分别反馈至恶意流量检测知识库与资源占用知识库。为管理这些数据，两个知识库创建多线程进程以接收各检测模块的反馈。图 5 展示了知识库监听端口的情况，其中显示了来自 7001 端口（对应 DRDoS 检测模块）的数据接收。

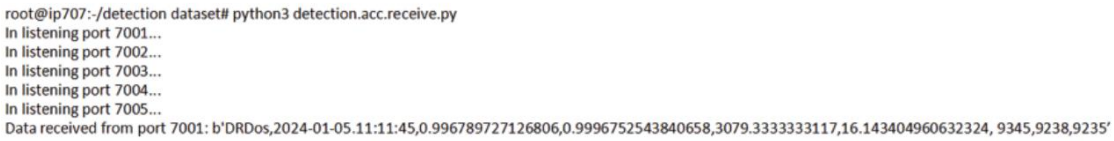

Figure 5 Monitoring of Knowledge Base

图 5 知识库的监听情况

图 6 显示，流量检测结果存储于恶意流量检测知识库的/root/detection_dataset/drdos-acc.csv，而承载检测模块的主机资源占用情况则存储于资源占用知识库的/root/detection_dataset/drdos-usage.csv。恶意流量检测结果中的时间戳已由 2023-12-29 更新为 2024-01-05，表明知识库中的恶意流量检测结果与系统资源占用情况均已完成更新。

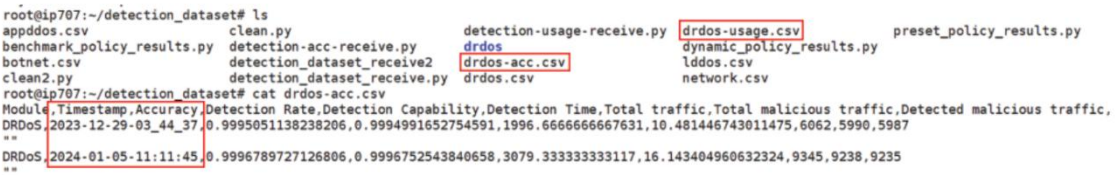

Figure 6 Update of Malicious Traffic Detection and Resource Usage Knowledge Base

图 6 恶意流量检测知识库与资源占用知识库的更新

流量特征信息（包括攻击流量的包速率、源 IP 熵、目的 IP 熵、源端口熵、目的端口熵、TTL 熵等包级特征）经 NETCONF 协议反馈至恶意行为知识库，该反馈过程如图 7 所示。

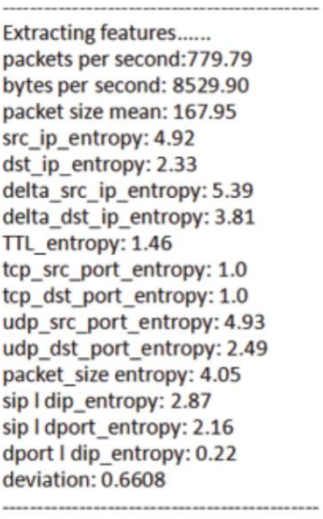

Figure 7 Knowledge Feedback

图 7 知识反馈

图 8 表明，流量的包级特征信息存储于恶意行为知识库的/monitor/monitor_information/feature。包含该信息的 CSV 文件已由 2023-06-12 更新为 2024-01-05，表明知识库中的包级流量特征已完成更新。

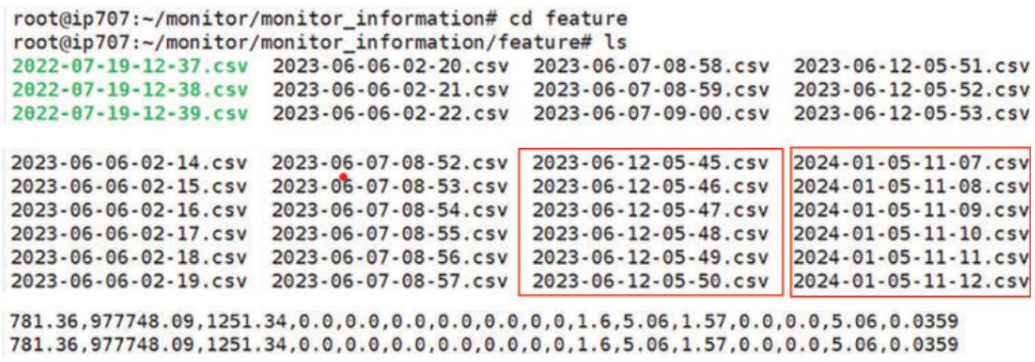

Figure 8 Update of Malicious Behavior Knowledge Base

图 8 恶意行为知识库的更新

（四）离线训练结果

本节包括特征选择模型与 SFC 路径选择模型的训练结果。基于 Python 工具，本节先介绍特征选择模型的参数调优与优化实现，并用已知数据集进行结果分析；随后给出 SFC 路径选择模型的训练结果，同样用已知数据集进行相应的结果分析。

1. 特征选择模型的训练结果

根据式（3），重要性得分随参数 α 与 β 变化。不同 α 与 β 取值下的得分：x 轴表示 83 种特征类型，y 轴表

示 α 的取值，z 轴表示重要性得分。本文把 α 从 0.1 按 0.1 的步长递增至 0.9，β 则反向变化。基于 RandomForest 的重要性得分波动显著，而 PageRank 的分布相对均匀，因此选取 α = 0.2 以确保得分不致过度受 RandomForest 影响。

为验证本文提出的 PageRank 与 RandomForest 相结合的特征选择算法的检测性能，本文生成了三组特征重要性排序，分别为 PageRank 与 RandomForest 结合、仅用 PageRank、以及所用的仅用 RandomForest。每组排序均取前 20 个特征，见表 5。

Table 5 Selected Features of Three Methods

表 5 三种方法所选出的特征

特征选择方法	特征名称
PageRank 与 RandomForest 结合	Fwd Packets/s, Flow IAT Mean, Flow Packets/s, Subflow Fwd Packets, Total Fwd Packet, Fwd Seg Size Min, Fwd IAT Std, Fwd Header Length, Fwd IAT Min, PSH Flag Count, Bwd Init Win Bytes, Flow IAT Std, Bwd Header Length, Flow IAT Min, FWD Init Win Bytes, Flow IAT Max, Fwd IAT Total, Subflow Fwd Bytes, Packet Length Max, Packet Length Variance
PageRank	Fwd Bulk Rate Avg, Fwd Packet/Bulk Avg, Average Packet Size, Bwd Segment Size Avg, Packet Length Min, Subflow Fwd Bytes, ACK Flag Count, Idle Min, PSH Flag Count, RST Flag Count, SYN Flag Count, FIN Flag Count, Packet Length Std, Packet Length Mean, Bwd Bytes Avg, Bwd Packet Avg, Bwd Bulk Rate Avg, Subflow Fwd Packets, Subflow Fwd Bytes, Subflow Bwd Packets
RandomForest	Fwd Packets/s, Flow IAT Mean, Flow Packets/s, Subflow Fwd Packets, Total Fwd Packet, Average Packet Size, Fwd IAT Std, Fwd Header Length, Subflow Fwd Bytes, PSH Flag Count, Bwd Init Win Bytes, Flow IAT Std, Bwd Header Length, Flow IAT Min, Packet Length Min, Flow IAT Max, Fwd IAT Total, Subflow Fwd Bytes, Packet Length Max, Packet Length Variance

基于表 5 列出的 20 个有效特征进行特征选择，并按 7∶3 的比例把数据集划分为训练集与测试集，详见表 6。DRDoS 特征数据集的样本总数为 1 299 286，其中训练样本 909 500 个、测试样本 389 786 个。在使用相同训练数据集的条件下比较各特征选择方法的性能。

Table 6 DRDoS Feature Datasets

表 6 DRDoS 特征数据集

数据集类型	正常流量样本数	攻击流量样本数
训练集	426 545	482 955
测试集	182 805	206 981

本文把测试数据集应用于 DRDoS 检测模块中的 XGBoost 并考察其输出结果。有三种不同特征选择方法所选特征对应的混淆矩阵。比较可见，PageRank 与 RandomForest 相结合的特征选择方法总体准确率高达 0.9999，在检测 Chargen、Memcached 与 TFTP 攻击时更取得了完全准确（1.0）的表现；相比之下，RandomForest 特征选择方法总体准确率为 0.9970，对 NTP 与 TFTP 攻击达到完全准确（1.0），但对 Chargen 攻击的准确率略低（约 0.9892）；而 PageRank 特征选择方法总体准确率为 0.9603，对 TFTP 攻击可完全识别（准确率 1.0），但对 Chargen

攻击的准确率较低。

基于三种特征选择方法的结果，本文进一步引入 LightGBM 与 KNN 两个机器学习模型，以及所讨论的 Stacking 集成学习模型，连同前述 XGBoost 检测模型，比较它们的训练时间与准确率。

三种特征选择结果在各模型上的训练时间。具体而言，采用 PageRank 与 RandomForest 特征选择的 KNN 模型、XGBoost 模型、LightGBM 模型与 Stacking 模型的训练时间分别为 75.81 s、160.09 s、73.10 s 与 364.67 s。这些数据一致表明，与其他特征选择方法相比，采用 PageRank 与 RandomForest 特征选择的模型训练时间最短。

三种特征选择结果在各模型上的准确率。结果表明，与另外两种特征选择方法相比，用 PageRank 与 RandomForest 组合选出的特征在每个训练模型中都表现出更高的准确率。该方法不仅检测时间更短，而且提升了对各类DRDoS攻击与正常流量的识别性能。实验结果凸显出本文特征选择方法能够提取与DDoS攻击更相关、影响力更大的特征，从而显著提升 NSF 的检测效率与检测性能。

2. SFC 路径选择模型的训练结果

在本文提出的 SFC 路径选择算法中，GAT+DRL 模型用 Pytorch 框架训练。本节给出训练 SFC 路径选择模型所用的相关参数与训练过程，以及用训练好的模型在已知攻击数据集上进行路径选择测试的结果。

实验用 Python 编码实现，训练过程中的相关参数设置详见表 7。

Table 7 Parameter Settings

表 7 参数设置

参数	取值
学习率	0.001
训练总周期	12 000
记忆库容量	10 000
探索因子	0.9
折扣因子	0.9
参数更新频率	200
优化器	Adam

GAT+DRL 模型在选定参数下、总计 12 000 次迭代的损失值与奖励值训练过程。损失值随训练迭代次数增加而稳步下降；路径的奖励值随训练迭代次数逐步上升，约在 5000 次迭代后收敛至最大值。

该模型在 19.4 min 的训练时长内达到收敛，并能够区分五种不同类型的 DDoS 攻击与混合攻击。

（五）在线测试结果

在线测试用训练好的模型对各类 DDoS 攻击流量实施实时检测，并向知识库模块提供反馈数据。本节旨在评估智能安全服务优化在提升系统检测性能与降低检测时间方面的效果，比较优化前后系统在精确率、召回率、F1 分数、恶意流量检测率、路径检测时延与 TMTDC 等指标上的表现。

系统中设置了一条包含全部 NSF 的预设路径，流量依次经过所有检测模块。本文回放了五类 DDoS 攻击流量，分别采用提出的 DQN 算法与本文提出的智能安全服务优化方法选择 SFC 路径，随后比较不同路径下的检测效果。表 8 给出了五类 DDoS 攻击流量在预设路径、DQN 算法选出的 SFC1 路径以及本文方法优化得到的 SFC2 路径下的 SFC 检测性能对比。例如对于 DRDoS 攻击，预设路径为['network', 'application', 'botnet', 'lddos', 'drdos', 'firewall']，DQN 算法选出的 SFC1 路径为['drdos', 'lddos', 'botnet', 'firewall']，而经本文优化方法选出的 SFC2 路径

为['drdos', 'firewall']。

Table 8 Comparison of Detection Performance under Different Paths

表 8 不同路径下检测性能的比较

攻击类型	所选路径	精确率	召回率	F1 分数	检测率
AppDDoS	预设路径	0.7653	0.7592	0.7892	0.7782
	SFC1 路径	0.8668	0.8798	0.8849	0.8841
	SFC2 路径	0.9785	0.9749	0.9667	0.9697
DRDoS	预设路径	0.8898	0.9778	0.9317	0.9331
	SFC1 路径	0.9292	0.9566	0.9574	0.9582
	SFC2 路径	0.9977	0.9943	0.9959	0.9991
NetDDoS	预设路径	0.9352	0.8924	0.9028	0.9028
	SFC1 路径	0.9813	0.9146	0.9492	0.9292
	SFC2 路径	0.9805	0.9922	0.9943	0.9925
LDDoS	预设路径	0.8778	0.8653	0.8898	0.8524
	SFC1 路径	0.9317	0.9428	0.9541	0.9510
	SFC2 路径	0.9934	0.9874	0.9757	0.9950
Botnet	预设路径	0.9232	0.9274	0.9268	0.9245
	SFC1 路径	0.9853	0.9425	0.9617	0.9646
	SFC2 路径	0.9613	0.9641	0.9711	0.9679

经智能安全服务优化后选出的 SFC2 路径，其精确率、召回率与 F1 分数在所有类型的 DDoS 攻击流量上均稳定超过 96%。与预设路径相比，SFC2 路径的恶意流量检测率平均提升 12.4%；与 DQN 方法所选路径相比平均提升 4.6%，总体表现优异。

智能安全服务优化前后，LDDoS、DRDDoS、NetDDoS 以及 AppDDoS 与 LDDoS 混合攻击等不同类型攻击流量的路径检测时延对比。攻击流量经预设路径时必须经过所有检测模块；而采用 DQN 方法或智能安全服务优化所选路径时，流量仅经过被选中的检测模块，因此时延相较预设路径显著降低。在本文的智能安全服务优化方法中，检测模块所选用的攻击流量特征与流量所经的 SFC 路径均被重新设计，与 DQN 方法所选路径相比，这既缩短了检测时间，也减少了各检测模块所遇到的 NSF 数量，从而降低了检测与转发带来的时延。因此，在所有类型的攻击流量下，本研究经智能安全服务优化所得路径的时延比 DQN 方法约低 42.8%，比预设路径约低 58.3%。

混合攻击流量条件下三条不同路径的 TMTDC：预设路径（蓝线）、DQN 方法所选路径（黄线）与经智能安全服务优化的路径（绿线）。预设路径的 TMTDC 约为 82，DQN 所选路径约为 87，而经智能安全服务优化的路径 TMTDC 显著更高，达到 97，比预设路径与 DQN 所选路径分别高出 18.1%与 11.5%，表明其检测能力更强。

（六）知识库更新前后的比较

为评估知识库更新后所选路径的有效性与智能化程度，本文先把流量通过 SFC 的路径设为预设路径，随后在 30 s 内对其施加 DRDoS 攻击。知识库更新前后的路径检测时延：起初预设路径的时延在 40 至 70 之间波动（紫色实线）；调整路径后，时延范围降至 20 至 40 之间（紫色虚线）。相较预设路径的这一时延下降，表明

知识库更新后所选路径的效率有所提升。

知识库更新前后的 TMTDC。攻击期间，预设路径的 TMTDC 始终低于 84（实线）；而在 15 s 后，自适应模型依据当前状况选出调整后的路径（虚线），其 TMTDC 显著提高至 97，比预设路径高出 15.5%。

综合分析可知，本文提出的智能安全服务优化方法具有优异的在线检测性能，对各类 DDoS 攻击的精确率、召回率与 F1 分数均稳定超过 96%；与预设路径和 DQN 所选路径相比，恶意流量检测率平均分别提升 12.4%与 4.6%，TMTDC 分别提升 18.1%与 11.5%，同时降低了检测时延。此外，该方法还能把检测结果、系统资源占用与网络流量包级特征反馈至相应知识库，为后续安全策略推理的更新提供依据。

五、结论

本文基于完备的网络安全知识库对智能安全服务进行优化，充分利用攻击流量特征、NSF 检测结果与系统资源占用等丰富的数据资源，并开发相应算法以增强安全服务的反馈能力、更新并推理知识库。对恶意流量的实时监测使知识得以持续更新，并实现快速、精准的闭环优化，从而支持路径策略的实时调整。实验结果表明，所提方案能够完成 SFC 路径选择，增强系统对 DDoS 攻击的检测能力，并有效降低流量穿越 SFC 的路径时延。

在未来工作中，本文将在真实网络环境中验证基于知识库的智能安全服务优化方法的有效性，并评估 GAT+DRL 模型中不同参数取值对路径选择的影响，以进一步提升路径的检测能力。

参考文献

[1] Mehmood A, Khanan A, Umar M M. Secure knowledge and cluster-based intrusion detection mechanism for smart wireless sensor networks. IEEE Access, 2017, 6: 5688-5694.

[2] 潘强. 智能无线传感器网络安全策略研究与实现. 沈阳师范大学, 2011.

[3] Zhang J, Wang Z, Ma N, et al. Enabling efficient service function chaining by integrating NFV and SDN: architecture, challenges and opportunities. IEEE Network, 2018, 32(6): 152-159.

[4] Mittal M, Kumar K, Behal S. Deep learning approaches for detecting DDoS attacks: a systematic review. Soft Computing, 2023, 27(18): 13039-13075.

[5] 郑承蔚, 王海凤, 刘瑞. SDN中DDoS攻击检测研究综述. 计算机工程与应用, 2024, 60(24).

[6] 王兴伟, 易波, 李福亮. SDN/NFV基本理论与服务编排技术应用实践. 北京: 科学出版社, 2021.

[7] Li K, Zhou H, Tu Z, et al. CSKB: a cyber security knowledge base based on knowledge graph. Singapore: Springer, 2020: 100-113.

[8] 朱若彬. 网络安全知识图谱构建关键技术研究. 哈尔滨工业大学, 2025.