基于深度学习的恶意代码检测算法设计与实现

张家华
北京鼎石国际学校，北京 101318
DOI:10.61369/ASDS.2026010009

摘　　要： 针对恶意代码检测中的特征表示不足与检测精度有限问题，提出一种基于多模态融合的深度学习检测方法。该方法首先从静态与动态两个维度设计并提取恶意代码的多模态特征表示，构建包含静态特征处理子网络与动态特征处理子网络的深度学习模型架构，通过特征融合模块实现跨模态信息的高效整合。进一步详细阐述模型的训练策略与优化细节，并完成检测系统的模块化实现与部署，实现了从样本预处理、特征提取、模型推理到结果可视化的全流程自动化。

关 键 词： 恶意代码检测；多模态特征；深度学习

Design and Implementation of Malicious Code Detection Algorithm Based on Deep Learning

Zhang Jiahua
Keystone Academy , Beijing 101318

Abstract: To address the issues of insufficient feature representation and limited detection accuracy in malicious code detection, a deep learning detection method based on multimodal fusion is proposed. This method first designs and extracts multimodal feature representations of malicious code from both static and dynamic dimensions, and constructs a deep learning model architecture that includes a static feature processing sub-network and a dynamic feature processing sub-network. Through a feature fusion module, efficient integration of cross-modal information is achieved. Further, the training strategy and optimization details of the model are elaborated in detail, and the detection system is modularly implemented and deployed, achieving full automation from sample preprocessing, feature extraction, model inference to result visualization.

Keywords: malicious code detection; multimodal features; deep learning

引言

随着网络攻击手段日趋复杂，恶意代码数量呈指数级增长，传统的基于特征代码与启发式规则的检测方法已经很难应对高层次持续威胁与未知变种。深度学习凭借其强大的特征学习能力和模式识别能力，给恶意代码检测带来了一场革命。通过对代码二进制结构、行为序列和可视化图像的分析，可以从海量数据中挖掘出深层的恶意模式，提高检测的准确性和泛化能力。本研究旨在系统设计并实现高效的深度学习检测算法，这对构建主动防御体系、保障关键信息基础设施安全具有重大理论价值与现实意义，是当前网络安全领域的迫切需求。

一、多模态恶意代码特征表示与提取设计

（一）总体设计思路

面对恶意代码种类繁多、规避技术不断演化，单模态特征描述已不能充分刻画恶意代码的恶意本质。本设计的核心思想是建立一种协同融合的多模态特征表达框架，从静态和动态两个正交互补的角度实现对恶意代码的全方位画像[1]。整体架构遵循“特征级融合”的先进思想，并不是简单的“拼接”，而是在高层抽象层次上对齐和整合。首先，需要建立统一的中间表示规范，保证不同模态特征能够在同一个向量空间中被度量和关联。本研究着重于特征的鲁棒性与解释性，引入注意力机制等深度学习模块，对不同模态、不同特征向量在判别任务中的贡献进行自动评估与权重，实现在保持原有语义信息的前提下提取最具判别性的恶意模式。

（二）静态特征表示设计

静态特征分析主要研究非运行状态下恶意代码的结构、语法和统计特性。其核心思想是突破传统基于规则的人工特征工程，采用深度学习模型对原始数据进行分层表达。对于可执行程序，如 PE，采用嵌入层或者一维卷积神经网络对二进制序列进行处

作者简介：张家华，男，汉族，辽宁鞍山人，研究方向：计算机科学。

 Copyright © This Work is Licensed under A Commons Attribution-Non Commercial 4.0 International License.

理，从而获得编码序列的语义和结构信息。同时，将文件反汇编得到的指令流视为一种文本序列，应用基于 Transformer 的预训练语言模型进行建模，以理解代码片段的功能语义。此外，将二进制文件可视化为灰度图像的方法被进一步深化，引入密集连接的卷积网络提取其全局与局部的纹理、形状及空间分布特征[2]。这些来自不同视角的静态表示将通过一个特征投影网络映射到共享子空间，形成统一的静态特征向量，其优势在于能够有效揭示代码的固有意图与潜在威胁，且分析效率极高。

（三）动态特征表示设计

动态特征表示旨在捕捉恶意代码在受控沙箱环境中运行时的行为语义，其设计重点在于对系统调用序列、API 调用图、网络流量、文件与注册表操作等运行轨迹进行结构化建模。将系统调用序列及其参数视为时间序列数据，采用门控循环单元或时序卷积网络捕获其长程依赖关系及顺序模式。针对较为复杂的应用程序接口依赖关系，将其抽象为有向图结构，利用图神经网络对节点和拓扑进行高层表示，揭示恶意行为的协同和触发机制。然后利用编码器将网络流量及行为记录转换成特征矢量[3]。动态特征提取的关键挑战在于行为轨迹的冗余与噪声，设计中引入了轨迹切片和关键行为筛选机制，并利用序列到序列的自动编码器进行降维与去噪，最终凝练出表征恶意代码核心动态行为的鲁棒特征向量。

二、多模态融合深度学习检测模型的设计

（一）模型总体架构设计

模型总体架构采用分层分流的混合深度学习框架，其核心是一个双通道并行处理网络与一个中心融合决策模块。该架构以特征级融合为主导策略，旨在实现动、静态信息的有效协同利用。静态特征处理子网络和动态特征处理子网络是两条独立的特征抽取骨干，分别从代码内在属性与运行时行为中学习深层的判别性表征[4]。每一个子网络中都包含了多个层次的非线性变换，它们负责将原始的多模态特征映射到高维的语义空间中。两个子网络输出分别输入到特征融合模块中，实现特征的拼接和加权求和，并利用跨通道注意机制和张量融合技术挖掘多个模态间的深层次关联和互补信息。

（二）静态特征处理子网络设计

静态特征处理子网需要同时处理多源异质数据，如代码语义、结构和可视化图像。对于运算码序列和反汇编文本，该子网络采用层次化的特征抽取策略。底层采用内嵌层将离散指令转换成密集矢量，再叠加一维卷积网络和门控环路单元层。卷积层用来捕捉本地的指令模式和语法结构，同时循环层对长距离的上下文依赖进行建模。针对二进制文件可视化图像，采用轻量级深度残差网络提取特征，通过残差连接有效缓解梯度丢失问题。采用全连通层对齐和降维，引入批规范化和 Dropout 层提高稳定性，防止过拟合。最后，在该子网络中输出一个具有统一、紧凑和信息量丰富的静态特征表达矢量。

（三）动态特征处理子网络设计

针对系统调用序列、API 调用流等时序数据，本网络以双向记忆网络为核心部件，通过双向结构实现对行为逻辑前后向的同步理解，并有效捕获恶意行为的因果关系。对于图结构的数据，如关系、进程树等，子网络与图卷积网络或者图注意力网络相结合。该网络利用消息传递机制聚集相邻节点的信息，学习其功能角色和结构重要性，实现非欧几何图数据向量表示的转换[5]。引入自关注层，自动评价不同时间阶段、不同行为事件在恶意判断中的作用权重，突出恶意行为关键片段。所有的动态特征在经过各自处理路径后，在融合层上聚合、压缩，得到动态特征向量，以反映行为语义。

（四）多模态特征融合模块设计

多模态特征融合技术突破了单纯的矢量拼接和元素叠加的局限，是实现模型性能提升的关键。该方法首先从静态和动态子网络中获取特征矢量，然后利用共享的全连通层将其映射到统一的特征空间。采用基于交叉注意理论的张量融合方法对核进行融合[6]。具体地，通过计算静态和动态特征间的交互注意权重矩阵，实现对不同维度特征交互强度和相关性的建模。然后，通过外积操作得到双模交互张量，将特征之间的高阶组合关系显式表示出来。这张数据经过多层感知机的压缩和提炼，提取出最有区别的跨通道协作特征。最后，在提取的协同特征基础上，采用小神经网络动态产生权重。

（五）模型训练细节设计

模型的训练遵循深度学习的标准范式，但针对多模态与安全领域的特殊性进行了精细化设计。该算法采用自适应矩估计方法，结合余弦退火学习速率调度策略，实现初始阶段的快速收敛和后期的精细调整。为避免过拟合问题，在网络结构中引入 Dropout、批量标准化等方法，同时针对多模态数据，研究多模态增强策略，如静态字节序列随机掩码、动态行为序列随机丢弃等，提高模型的鲁棒性。在训练过程中，采用分步学习的策略，对动、静两个子网络分别进行独立的预训练，并在训练过程中引入辅助标记，使其特征抽取能力更加稳定。通过端到端联合训练，对网络参数尤其是融合模块的权值进行调整。利用检验集进行早停监测与超参调整，保证模型对未知数据保持最优的泛化性能。

三、检测系统的实现与部署

（一）系统需求分析与整体架构

该系统旨在实现一个面向高吞吐量、低延迟环境的自动化恶意代码检测平台。核心需求包括支持对 PE、ELF、文档宏等多种文件格式的自动化分析；具有静、动态双方面的分析技能；将训练好的多模态深度学习模型用于实时或批处理；提供交互式测试报告。整个体系结构采用了 Microservice 设计模式，保证了模块的松耦合性、高内聚性和独立的可伸缩性。系统分为四层：数据录入与调度层，核心分析引擎层，模型服务层和用户界面层。数据输入层负责接收原始样本文件，将样本文件分发到静态和动态分析队列中。核心分析引擎由两个独立的服务（样本预处理和特征抽取）组成，分别部署在独立的计算环境中。模型服务层采用容器技术对深度学习模型进行封装，并提供高性能 REST 式 API

Copyright © This Work is Licensed under A Commons Attribution-Non Commercial 4.0 International License.

接口。用户界面层提供带有结果可视化仪表盘的网络管理接口。各层采用消息队列和 API 网关通讯，保证了系统的异步处理能力，可靠，可扩展性强。

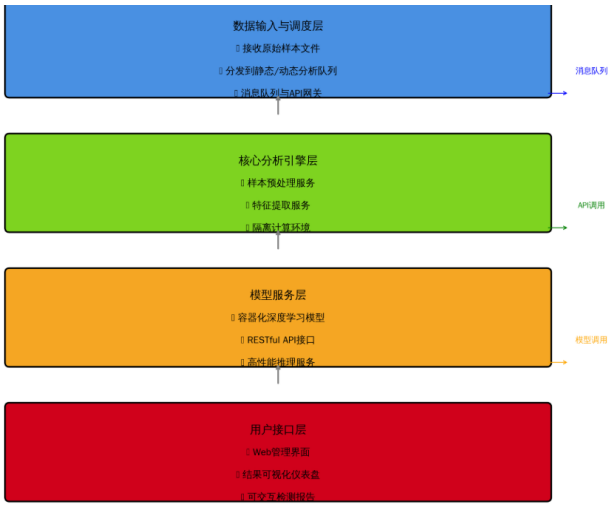

图1：整体架构图

（二）核心模块实现

1. 样本预处理模块实现

样本预处理模块主要负责接收原始文档，对原始文档进行初步的标准化和分流。在实现过程中，首先利用文件的魔数、扩展名等信息对文件格式进行快速地识别，并对文件的完整性进行检查。针对可执行程序，利用开源 LIEF 库对其进行解析，提取结构元数据，如文件头信息，输入导出表等，并对其进行初步识别和记录。然后，这个模块会启动一个轻量级的静态分析子过程，生成带有控制流图的拆装器代码。同时，对于需要进行动态分析的样本，该模块可以自动生成适用于沙箱或自定义沙盒环境的概要文件，并将其提交到动态分析任务队列中。通过前处理流程设计成流水式流程，对关键步骤进行日志记录，生成统一的样本元数据 JSON 描述文件，为后续特征抽取提供规范的输入接口和上下文信息。

2. 特征提取模块实现

基于多模态特征表达，特征抽取模块设计了静态和动态两个独立的处理流程。静态特征管线从预处理模块输出的解编文本、二进制字节和结构信息进行处理。将操作代码序列和 API 调用序列数字化，生成相应的嵌入矢量。可视化影像的产生是通过把二进制文件重新塑造成具有一定宽度的灰阶影像矩阵。动态特性管道对沙盒的执行环境进行监控，捕捉系统调用顺序，网络活动、注册表、文件操作等行为。该系统调用序列通过进程和线程 ID 进行会话划分与清洗。图结构特征从进程树和 API 调用依赖中构建邻接矩阵。所有原始特征经过标准化后，由相应的特征提取器转化为预设维度的特征向量。该模块的核心在于一个特征注册与管理中心，它统一调度各类特征提取器，确保特征生成的一致性和可复现性，并将最终的多模态特征向量存入高性能的向量数据库中以供模型推理调用。

3. 深度学习模型推理模块实现

该服务采用高性能深度学习推理引擎对模型进行加载和优化。服务显示从特征抽取模块接收特征矢量的统一 gRPC 或 REST 风格 API 端点。在界面内，服务首先验证输入的静态和动态特性矢量，并对其进行维度校准。然后调用优化后的模型计算图用于正向传输。推理过程充分利用了 GPU 强大的并行运算能力，并通过批量处理技术提高了系统的吞吐率 [7]。为了保证系统的稳定性和性能，系统采用了动态批处理机制、请求队列管理机制和自动扩展机制。对推理请求和结果进行了详细的日志记录，并将结果反馈给系统的监测部件。

4. 结果生成与可视化模块实现

结果生成和可视化模块完成了预处理、特征提取和模型推理等各个环节的融合，生成结构化的检测报告，并提供可视化的分析界面。首先将模型输出的原始概率得分与显著性分析结果相结合，利用可配置的判决引擎，将恶意判断标签和置信度转换为最终的恶意判断标签和置信度。报表产生器会汇编样本元数据，静态和动态分析总结，关键的恶意特性度量和模型推断细节，并以结构化的 JSON 或者 HTML 格式输出 [8]。可视化界面则基于 Web 技术开发，提供交互式图表展示样本的行为序列图、API 调用关系图以及模型关注度的热力图。此外，该模块维护一个历史检测数据库，支持对检测结果的聚合统计、趋势分析与关联挖掘。

表1：可视化模块关键输出数据

字段名称	数据类型	描述
Sample_ID	String	样本唯一标识符 (MD5/SHA256)
Final_Verdict	String	最终判定结果（恶意 / 良性）
Confidence_Score	Float	模型判定置信度 (0–1)
Top_Static_Features	Array[String]	权重最高的静态特征描述列表
Top_Dynamic_Behaviors	Array[String]	权重最高的动态行为描述列表
Report_Path	String	完整分析报告存储路径
Visualization_URL	String	交互式可视化界面访问链接

（三）系统集成与部署

采用容器和编排技术实现了系统的集成，所有的核心模块都以独立的容器镜像形式封装，并由 Kubernetes 统一编排、部署和管理。通过配置映射和安全对象来管理具有密钥的系统配置信息。数据持久层采用高性能分布式文件系统对样本文件进行存储；消息队列是整个系统的中枢，它将各个微服务连接在一起，实现异步解耦。整个部署体系结构采用前、后端分离的方式，前端网络接口通过 API 网关和后端服务集群进行通讯。该系统具有完备的监测报警系统，利用 Prometheus 采集系统的性能指标，通过 Grafana 在仪表盘上展示，实时监测服务健康状况、资源利用率和监测业务指标。

表2：系统部署资源配置

服务组件	容器副本数	CPU 资源请求 / 限制	内存资源请求 / 限制	存储卷挂载	网络策略
预处理调度器	2	0.5 / 2 核	1Gi / 4Gi	无	内部服务
静态特征提取器	3	1 / 4 核	2Gi / 8Gi	样本缓存卷	内部服务
动态沙箱集群	5	2 / 8 核	4Gi / 16Gi	隔离数据卷	严格出站
模型推理服务	2	申请 GPU	4Gi / 16Gi	模型存储卷	内部服务

 | Copyright © This Work is Licensed under A Commons Attribution-Non Commercial 4.0 International License.

消息队列	3	0.5 / 1 核	512Mi / 2Gi	持久化数据卷	内部服务
API 网关与 Web 前端	2	0.25 / 1 核	256Mi / 1Gi	无	面向互联网

四、结语

本研究设计并实现了一种基于多模态融合深度学习的恶意代码检测方案。构建静态和动态双模态特征表达体系，设计特征处理子网和融合模块，提高模型的识别准确率和泛化能力。通过系统的集成和部署，证明了所提出的方法是可行和实用的。未来工作将集中于引入更丰富的代码表征模态、探索更高效的自适应融合机制，并研究模型在对抗性样本攻击下的鲁棒性增强方法，以应对日益复杂和隐蔽的恶意代码威胁。

参考文献

[1] 李梦，刘万平，黄东 . 基于特征融合的恶意代码检测 [J]. 计算机工程与设计 ,2024,45(12):3568-3574.

[2] 蒋应瑞，黎秋玲 . 一种基于卷积神经网络的恶意代码检测模型 [J]. 江苏通信 ,2024,40(06):102-105.

[3] 熊其冰，苗启广，杨天，等 . 一种基于混合量子卷积神经网络的恶意代码检测方法 [J]. 计算机科学 ,2025,52(03):385-390.

[4] 蒲经纬，张辉，唐斌 . 基于深度学习的恶意代码检测技术研究 [J]. 网络安全技术与应用 ,2024,(10):39-43.

[5] 宋亚飞，张丹丹，王坚，等 . 基于深度学习的恶意代码检测综述 [J]. 空军工程大学学报 ,2024,25(04):94-106.

[6] 张晓良，柴艳玉，吴克河，等 . 一种基于增量学习的恶意代码检测方法 [J]. 计算机与数字工程 ,2024,52(07):2141-2145+2220.

[7] 靳黎忠，薛慧琴，段明博，等 . 基于多频特征学习的恶意代码变种分类 [J]. 计算机工程与设计 ,2024,45(07):1934-1940.

[8] 尚承翔，李桦宇，李瀚洋，等 . 基于深度迁移学习的恶意代码可视化检测 [J]. 网络安全技术与应用 ,2024,(03):37-39.

Copyright © This Work is Licensed under A Commons Attribution-Non Commercial 4.0 International License.