doi:10.16018/j.cnki.cn32-1650/n.202601003

基于混合自编码器模型的XSS攻击检测研究

赵润钰[1],赵生慧[2]
(1. 安徽理工大学 计算机科学与工程学院, 安徽 淮南 232000;
2. 滁州学院 人工智能学院, 安徽 滁州 239000)

摘要:针对传统检测方法,包括基于规则的方法和传统深度学习模型难以适应新兴攻击技术,难以有效捕捉跨站脚本(Cross-site scripting, XSS)攻击载荷细微差异与上下文相关性的局限,提出了一种融合检测模型UBL-XSS。该模型创新性地结合了BERT、通用句子编码器和基于LSTM的自编码器3种深度学习技术,从多个维度捕获XSS攻击载荷的语义和句法特征,实现了基准数据集上99.95%的精确度和99.84%的召回率,显著优于文中提出的现有主要检测方法。

关键词:XSS攻击检测;自编码器;BERT;USE
中图分类号:TP393.08 **文献标志码**:A **文章编号**:1671-5322(2026)01-0017-06

Research on XSS Attack Detection Based on Hybrid Autoencoder Models

ZHAO Runyu[1], ZHAO Shenghui[2]
(1. School of Computer Science and Engineering, Anhui University of Science and Technology, Huainan Anhui 232000, China;
2. School of Artificial Intelligence, Chuzhou University, Chuzhou Anhui 239000, China)

Abstract: To address the limitations of traditional detection methods, including rule-based approaches and conventional deep learning models, which struggle to adapt to emerging attack techniques and fail to effectively capture the subtle differences and contextual dependencies of cross-site scripting(XSS) payloads, this paper proposes a fusion detection model named UBL-XSS. The model innovatively integrates three deep learning techniques including BERT, universal sentence encoder, and LSTM-based autoencoders, to capture the semantic and syntactic features of XSS payloads from multiple dimensions. Experimental results demonstrate that UBL-XSS achieves a precision of 99.95% and a recall of 99.84% on benchmark datasets, significantly outperforming the state-of-the-art detection methods discussed in this study.

Keywords: XSS attack detection; autoencoder; BERT; USE

随着互联网技术的迅速发展,网络安全越来越引起人们的重视。在Web应用领域,跨站脚本(cross-site scripting,XSS)攻击是当前最普遍且危险的攻击方式之一,其核心机制在于攻击者成功地将恶意脚本注入到终端用户所信任的网站,让该脚本在用户浏览器上运行,进而窃取用户敏感信息或劫持用户会话[1]。

传统的XSS攻击检测主要依赖于静态分析[2]和动态分析[3],其中静态分析因无法捕获脚本动态行为,可能会忽略某些类型的XSS攻击;而动态

收稿日期:2025-12-24
基金项目:滁州市委网信办横向项目(HX2022122)。
作者简介:赵润钰(1998—),女,安徽六安人,硕士生,主要研究方向为Web安全。
通信作者:赵生慧(1970—),女,安徽滁州人,教授,博士,主要研究方向为智能信息处理、物联网技术与应用。

分析严重依赖于实际执行,可能导致更高的计算开销和复杂攻击的潜在规避。为了克服这些限制,研究人员越来越多地转向基于机器学习(machine learning,ML)的方法[4],包括决策树、随机森林、支持向量机、逻辑回归等,并取得了不同程度的成功[5-6]。例如,Zhou等[7]利用贝叶斯网络和威胁情报进行检测,在恶意记录占比45%的测试场景下仍能达到98.54%的准确率。

近年来,深度学习(deep learning,DL)进一步推动了XSS攻击检测的发展[8]。Krishnan等[9]结合ML模型和深度神经网络模型,并采用URL编码与字典映射机制,提出了一种混合堆叠集成学习方法,在多个数据集上实现了超过99.5%的检测准确率;Lei等[10]研究发现长短期记忆神经网络(long short-term memory networks, LSTM)在检测复杂XSS载荷时,精确度和召回率均优于传统机器学习方法。

随着人工智能技术的发展,通过特征建模和行为分析,人工智能技术能够有效识别潜在的恶意脚本,显著提升检测的自动化水平和准确性[11]。Peng等[12]提出基于Transformer的XSS检测方法,通过自动生成变形攻击向量解决数据集单一及陈旧问题,效果显著优于传统的深度学习模型。基于Transformer的架构,特别是双向编码器表征法(bidirectional encoder representations from transformers, BERT)的出现,在最近的一些研究中已将它应用于网络钓鱼检测[13]和SQL注入识别[14]等,但BERT在XSS攻击检测中的应用尚未得到充分探索。

本文在上述文献的基础上,构建了一个多维异构特征空间,通过同时捕获细粒度的词语语义和更广泛的句子上下文,来弥补单一模型的局限性,从而增强XSS攻击载荷的特征提取能力,有效缓解在面对新型攻击变体时出现的性能下降问题。具体方法如下:

1)利用通用句子编码器(universal sentence encoder,USE)在XSS攻击检测领域提供的独特优势,将文本数据转换为捕获语义相似性和含义的高维向量[15],从而挖掘整个Web内容的潜在意图,更有效地识别潜在的XSS攻击;

2)利用BERT的双向编码器结构精准捕获深层句法和复杂的语义依赖关系,从而产生高质量的特征表示,显著提高检测的准确性和泛化能力;

3)采用基于LSTM的自编码器(LSTM-based autoencoder, LSTM-AE)模型作为无监督ML模型,通过识别异常模式来检测网络流量中的未知攻击。

本文的主要贡献如下:1)将LSTM-AE模型的无监督学习机制与基于句子的多维嵌入技术相结合,开发了一种专门为XSS攻击检测设计的混合DL模型,使系统在适应演变攻击策略和识别新型混淆技术的同时,提升整体检测性能;2)整合3种DL架构的句子级嵌入表示,提出了一种经消融研究验证的嵌入融合机制,在基准数据集上实现了99.95%的精确度和99.84%的召回率;3)设计了基于分类结果的反馈循环,对自编码器进行无监督的"自我纠正",以弥补单一模型的局限性。

1 UBL-XSS混合模型

1.1 数据预处理

本文开发的专门为XSS攻击检测设计的混合DL模型为UBL-XSS混合模型,该模型采用3种基于句子的技术,融合了LSTM自动编码器、BERT双向编码器和USE编码器的互补优势,实现对XSS攻击的高效检测。

UBL-XSS混合模型结构如图1所示。该模型在对XSS攻击进行检测前,首先要对检测代码片段进行预处理,具体处理方法如下:

1)针对待检测代码片段执行解码操作,以识别其恶意或良性特征。攻击者常采用混淆策略规避传统过滤与验证机制的检测,这些混淆手段包括Unicode转换、十六进制编码、Base64加密及UTF-7编码等多种技术。本研究提出的解码模块能够穷举各类编码可能性,将经过混淆的代码还原至标准格式。

2)对经过解码操作形成标准格式的代码进行规范化处理,旨在消除解码后数据中的冗余信息与噪声。具体规范化操作包括:剔除空格、过滤特殊符号、移除协议前缀"http://",并将函数参数统一转换为参数字符串标识"param_strings"。

3)对数据进行分词处理,目的是将字符串序列分解成包含输入字符、子字符或子组的片段,从而最小化数据长度并降低复杂性,进而减少了数据处理的成本。通过分词保留了代码片段中重要的词,提高了检测的准确性。

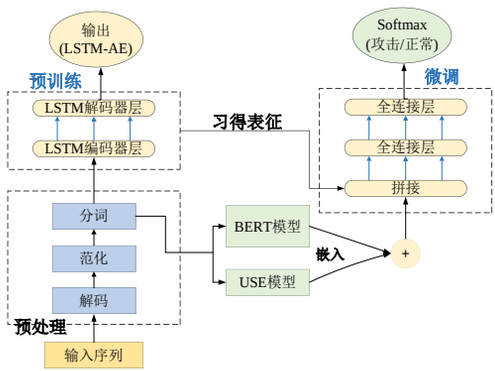


图 1　UBL-XSS 模型结构图

Fig. 1　UBL-XSS model architecture

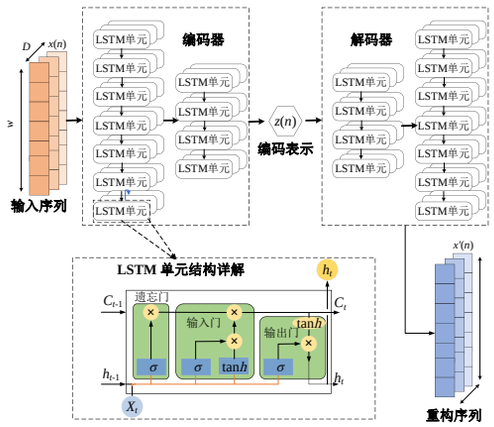


图 2　LSTM-AE 模型的网络结构图

Fig. 2　LSTM-AE model architecture

1.2　UBL-XSS 检测模型构建

经过预处理后的数据集需要通过自编码器进行预训练，以提升对异常结构的感知能力。自编码器是一种无监督的神经网络，通过最小化重构误差，从数据集中学习最佳的编码-解码方案。为确保模型学习到高阶特征表示，瓶颈层的维度应显著小于输入维度，从而实现对原始载荷的压缩与关键模式提取。

为了提升训练效果，本文融合了 LSTM 自动编码器（LSTM-AE 编码器）、BERT 模型和 USE 模型的互补优势，提出了一种经消融研究验证的嵌入融合机制。该机制使用标准交叉熵损失函数通过测量预测类别与真实标签之间的差异进行训练，计算公式如下：

$$L = -\sum_{i=1}^{N}\left(y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i)\right) \quad (1)$$

式中：L 代表标准交叉熵损失函数；$y_i \in \{0, 1\}$ 表示第 i 个样本的实况标签；$\hat{y}_i$ 表示恶意类别的预测概率；N 代表样本总数。

通过最小化该损失函数，模型学会了有效区分正常输入和攻击输入。这种架构不仅能够捕获表面级词汇特征，还对深层句法和语义模式进行建模，使其能够抵御 XSS 攻击中普遍存在的对抗技术，如字符编码、脚本注入和标签操作。

1.2.1　LSTM-AE 模型

LSTM-AE 编码器结构如图 2 所示。在 XSS 攻击检测场景中，LSTM-AE 编码器通过学习 XSS 有效负载和正常网络请求的潜在表示，能够识别异常模式并提取具有判别能力的特征。对于任意输入样本 X，编码器提取其潜在表示的公式如下：

$$z_{\mathrm{AE}} = \mathrm{Encoder}(X) \in \mathrm{R}^{128} \quad (2)$$

式中：z_{AE} 为编码器提取的 128 维特征。

由于模型学习了正常请求和攻击的潜在表示差异，编码器提取的特征包含了“偏离正常模式”的信息，因此即使对于训练时未见过的新型 XSS 攻击变种，编码器仍能生成异常的特征向量，为后续分类器提供判别信号。

1.2.2　通用句子编码器

USE 是一个强大的自然语言处理框架，能够将不定长文本输入转换为固定维度的向量表示，从而编码语义信息和上下文特征[16]。该架构采用在大规模文本语料上训练的深度神经网络，生成高保真度的嵌入向量以捕获语义关系和句间相似性。USE 已在多种 NLP 任务中表现出色，包括文本分类、语义相似度度量和情感分析。本研究将 USE 作为 XSS 攻击检测的特征提取机制，利用其生成 Web 内容有意义表示的能力，从而提升检测模型性能。数学上，句子 s 的 USE 嵌入 z_{USE} 表示如下：

$$z_{\mathrm{USE}} = f_{\mathrm{USE}}(s) \in \mathrm{R}^{m} \quad (3)$$

式中：m 是 USE 嵌入的维度，大小为 512。

1.2.3　BERT

BERT 利用多头自注意力机制和深度双向编码来捕获自然语言序列中的复杂依赖关系[17]，其架构如图 3 所示。该模型以预处理阶段生成的 token IDs、注意力掩码和段 IDs 序列三元组作为输入，将它们转换为 128 维的上下文隐藏状态序列，转换公式如下：

$$\boldsymbol{H}_i = \mathrm{BERT}(I_i, A_i, S_i) = [h_1, h_2, \cdots, h_{128}] \quad (4)$$

式中：I_i、A_i、S_i 分别表示预处理阶段生成的 token IDs、注意力掩码和段 IDs；$\boldsymbol{H}_i$ 表示输入序列中由 128 个数据组成的每个 token 的隐藏状态向量。

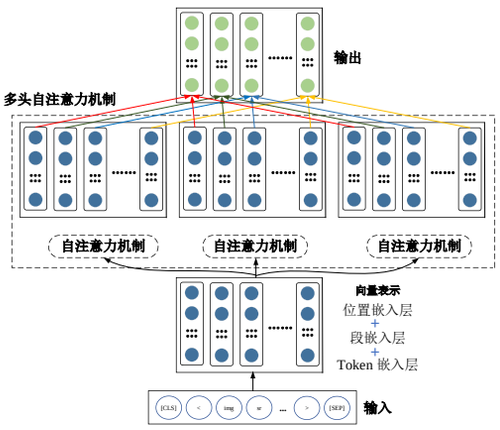


图 3　BERT 模型架构

Fig. 3　BERT model architecture

对应于特殊分类 token[CLS]的向量，记为 z_{BERT}，作为 BERT 模型输出的第一个位置的隐藏状态向量，如公式(5)所示。

$$z_{\mathrm{BERT}} = \boldsymbol{H}_i[0] \in \mathrm{R}^{d_{\mathrm{bert}}} \tag{5}$$

式中：d_{bert} 表示 BERT 模型的隐藏层维度。

向量 z_{BERT} 聚合了整个输入序列的全局语义信息，作为下游分类任务的句子级表示。

1.3　特征拼接与分类

根据图 1 的流程，将 BERT 和 USE 分别生成的嵌入进行融合，再与 LSTM 自编码器中编码器提取的特征进行连接，得到新的用于 XSS 攻击分类的特征向量 z_{fused}。连接公式如下：

$$z_{\mathrm{fused}} = [z_{\mathrm{BERT}}, z_{\mathrm{USE}}, z_{\mathrm{AE}}] \in \mathrm{R} \tag{6}$$

将式(6)中得到的融合特征向量输入到全连接层进行特征向量学习后，再传递到 Softmax 进行分类，用于正常 Web 请求或 XSS 攻击的最终分类，计算公式如(7)所示。

$$\hat{y} = \mathrm{soft\,max}(W \cdot z_{\mathrm{fused}} + b) \tag{7}$$

式中：W 和 b 分别代表权重和偏置；输出 $\hat{y} \in \mathrm{R}^2$ 包含输入分别属于正常、恶意类别的预测概率。

2　实验与分析

2.1　实验环境与数据集

为评估 UBL-XSS 检测模型的有效性，本文采用的实验环境为 Ubuntu 22.04 操作系统、Intel® Xeon® Platinum 8352V CPU@2.10 GHz 处理器、90 GB RAM 内存、NVIDIA vGPU-32 GB GPU、Python 3.12 编程语言、PyTorch 2.5.1DL 框架、CUDA 12.4 并行计算环境，数据集来自文献[18]。该数据集中包含约 20 万条正常网络请求的负样本、4 万条恶意攻击请求的正样本，样本类别用[0,1]表示，其中 0 代表良性，1 代表恶意。

实验时，将正样本和负样本混合后，按照 8∶1∶1 的比例随机划分为训练集、验证集和测试集，其中训练集用于模型的参数学习，验证集用于模型的超参数优化与调整，而测试集则用于模型的最终性能评估。

2.2　评价指标

本文采用准确率 A、精确度 P、召回率 R 和 F_1 值作为评价指标来评估模型的性能，各指标计算公式如下：

$$A = \frac{T_{\mathrm{P}} + T_{\mathrm{N}}}{T_{\mathrm{P}} + T_{\mathrm{N}} + F_{\mathrm{P}} + F_{\mathrm{N}}} \times 100\% \tag{8}$$

$$P = \frac{T_{\mathrm{P}}}{T_{\mathrm{P}} + F_{\mathrm{P}}} \times 100\% \tag{9}$$

$$R = \frac{T_{\mathrm{P}}}{T_{\mathrm{P}} + F_{\mathrm{N}}} \times 100\% \tag{10}$$

$$F_1 = \frac{2 \times P \times R}{P + R} \times 100\% \tag{11}$$

式中：T_{P}、T_{N}、F_{P}、F_{N} 分别表示真正例、真负例、假正例、假负例。

2.3　模型参数设置

在构建混合 DL 模型时，涉及多个参数的设置与优化。通过实验过程反复调试各项参数，最终使模型性能达到最佳状态，此时各参数配置如表 1 所示。

表 1　主要实验参数设置

Table 1　Main experimental parameter settings

实验参数	参数值
训练轮数	20
LSTM 自编码器训练轮数	50
批处理大小	32
输入词向量	BERT
LSTM 隐藏层大小	256
学习率	2×10^{-5}
LSTM 自编码器学习率	1×10^{-3}
dropout 层	0.5
损失函数	交叉熵
优化函数	Adam

2.4　消融实验

为了验证 UBL-XSS 混合模型的有效性，通过消融实验对来自于文献[18]的数据集进行验证，得到各指标的结果如表 2 所示。由表 2 可知，仅使用 LSTM-AE 效果不佳，4 项指标均最低；使用 BERT 与 LSTM 自编码器的组合即 BERT-LSTM 自

表2　消融实验中各模块及其组合的性能对比
Table 2　Ablation results of various modules and their combinations　%

模块	A	P	R	F_1
LSTM编码器	96.42	95.88	94.12	94.99
USE-LSTM编码器	99.52	99.38	98.24	98.81
BERT-LSTM编码器	99.84	99.81	99.15	99.48
UBL-XSS	99.96	99.95	99.84	99.89

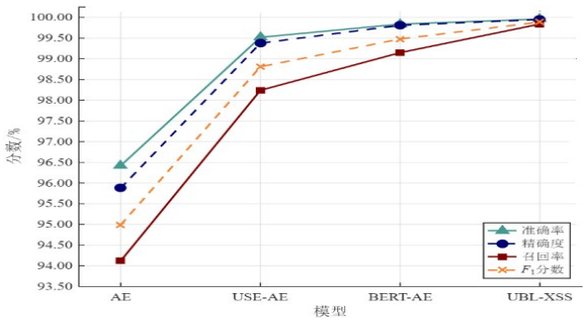


图4　模型消融实验
Fig. 4　Model ablation study

编码器方案，4项指标均较USE与LSTM自编码器的组合即USE-LSTM自编码器方案高，但仍然比集成三者输出方案的UBL-XSS混合模型低。

为直观展示消融实验中各模块对模型性能的影响，根据表2的实验结果绘制了不同配置下各评价指标的对比图，如图4所示。

2.5　对比实验

为验证本文所提UBL-XSS混合模型的优越性，对XSS攻击检测的几种常用方法在来自于文献[18]的数据集上开展对比测试，结果如表3所示。表3的实验结果表明，UBL-XSS混合模型在所有性能指标上均表现最佳，这可能是由于：1）UBL-XSS混合模型融合了3种不同的句子嵌入表示方法，能够从多个维度捕获XSS攻击载荷的语义和句法特征；2）LSTM-AE通过无监督学习识别异常模式，使系统能够适应演变的攻击策略并检测新型混淆技术；3）优化的嵌入融合策略能够同时捕获局部句法模式和全局语义上下文。3个模型协同互补，使得本文提出的UBL-XSS混合模型在准确率、精确度、召回率以及F_1分数上得到了进一步提升。

为便于清晰展示本文方法相对于其他深度学习方法在各个评价指标上的性能表现，依据表3实验结果绘制了对比图，如图5所示。

表3　常用的XSS攻击检测方法的效果比较
Table 3　Performance comparison of common XSS attack detection methods　%

模型	A	P	R	F_1
DeepXSS[19]	99.27	98.65	97.14	97.89
DCNN-GRU[20]	99.60	99.88	97.76	98.81
DCNN-Transformer	99.68	99.72	98.35	99.03
Transformer-LSTM	99.74	99.70	98.84	99.27
BERT	99.78	99.75	98.97	99.36
BERT-BiLSTM	99.79	99.80	99.09	99.45
BERT-TextCNN	99.64	98.42	99.49	98.95
UBL-XSS	99.96	99.95	99.84	99.89

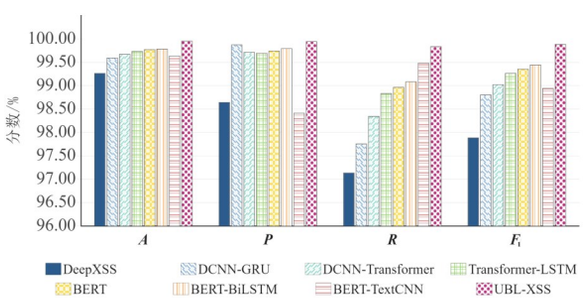


图5　性能对比
Fig. 5　Performance comparison

3　结论

针对XSS攻击载荷的复杂性与混淆性，提出了一种多维特征融合的检测模型UBL-XSS。该模型创新性地整合了BERT、USE以及LSTM-AutoEncoder模块，构建了多层次的句子级嵌入架构；通过全连接层学习来自每种技术的嵌入，在预测分类概率后传递给Softmax进行最终分类。实验结果表明，UBL-XSS在所有性能指标上均表现最佳，显著优于现有主流方案。

参考文献：

[1] SHARIF M H U. Web attacks analysis and mitigation techniques[J]. International Journal of Engineering Research & Technology (IJERT), 2022: 10-12.

[2] USHA G, KANNIMUTHU S, MAHENDIRAN P D, et al. Static analysis method for detecting cross site scripting vulnerabilities[J]. International Journal of Information and Computer Security, 2020, 13(1): 32-47.

[3] 王铃铜, 王慧玲, 徐苗, 等. 跨站脚本攻击检测与防御技术综述[J]. 计算机应用研究, 2024, 41(3): 652-662.
[4] KAUR J, GARG U, BATHLA G. Detection of cross-site scripting (XSS) attacks using machine learning techniques: a review[J]. Artificial Intelligence Review, 2023, 56(11): 12725-12769.
[5] THAJEEL I K, SAMSUDIN K, HASHIM S J, et al. Machine and deep learning-based XSS detection approaches: a systematic literature review [J]. Journal of King Saud University: Computer and Information Sciences, 2023, 35(7): 101628.
[6] ALHAMYANI R, ALSHAMMARI M. Machine learning-driven detection of cross-site scripting attacks[J]. Information, 2024, 15(7): 420.
[7] ZHOU Y, WANG P C. An ensemble learning approach for XSS attack detection with domain knowledge and threat intelligence[J]. Computers & Security, 2019, 82: 261-269.
[8] TADHANI J R, VEKARIYA V, SORATHIYA V, et al. Securing web applications against XSS and SQLi attacks using a novel deep learning approach[J]. Scientific Reports, 2024, 14: 1803.
[9] KRISHNAN M, LIM Y, PERUMAL S, et al. Detection and defending the XSS attack using novel hybrid stacking ensemble learning-based DNN approach[J]. Digital Communications and Networks, 2024, 10(3): 716-727.
[10] LEI L, CHEN M, HE C, et al. XSS detection technology based on LSTM-attention[C]//2020 5th International Conference on Control, Robotics and Cybernetics (CRC). October 16-18, 2020, Wuhan, China. IEEE, 2020: 175-180.
[11] SOHAIL A, AYISHA B, HAMEED I, et al. Deep neural networks based meta-learning for network intrusion detection [J]. arXiv preprint arXiv:2302. 09394, 2023.
[12] PENG B T, XIAO X Y, WANG J. Cross-site scripting attack detection method based on transformer[C]//2022 IEEE 8th International Conference on Computer and Communications (ICCC). December 9-12, 2022, Chengdu, China. IEEE, 2023: 1651-1655.
[13] RIFAT N, AHSAN M, CHOWDHURY M, et al. BERT against social engineering attack: phishing text detection[C]// 2022 IEEE International Conference on Electro Information Technology (EIT). May 19-21, 2022, Mankato, MN, USA. IEEE, 2022: 1-6.
[14] LODHA S, GUNDAWAR A. SQL injection and its detection using machine learning algorithms and BERT[C]//International Conference on Cognitive Computing and Cyber Physical Systems. November 26-27, 2022, Bhimavaram, Andhra Pradesh. Springer Nature Switzerland, 2022: 3-16.
[15] AL-SMADI M, HAMMAD M M, AL-ZBOON S A, et al. Gated recurrent unit with multilingual universal sentence encoder for Arabic aspect-based sentiment analysis[J]. Knowledge-Based Systems, 2023, 261: 107540.
[16] FATTOH I E, KAMAL ALSHEREF F, EAD W M, et al. Semantic sentiment classification for COVID-19 tweets using universal sentence encoder[J]. Computational Intelligence and Neuroscience, 2022, 2022: 6354543.
[17] GARDAZI N M, DAUD A, MALIK M K, et al. BERT applications in natural language processing: a review[J]. Artificial Intelligence Review, 2025, 58(6): 166.
[18] 赵斯祺, 代红, 王伟. 基于Transformer-LSTM模型的跨站脚本检测方法[J]. 计算机应用与软件, 2023, 40(9): 327-333
[19] FANG Y, LI Y,LIU L. DeepXSS: Cross site scripting detection based on deep learning[C]//Proceedings of the 2018 International Conference on Computing and Artificial Intelligence. March 12-14, 2018, Chengdu, China. ACM, 2018: 47-51.
[20] LUO J, XU G Q. XSS attack detection methods based on XLNet and GRU[C]//2021 4th International Conference on Robotics, Control and Automation Engineering (RCAE). November 4-6, 2021, Wuhan, China. IEEE, 2021: 171-175.

(责任编辑:李华云)