基于联邦大模型的网络攻击检测方法研究

康海燕[1,2],张义钒[1,2],王楠敏[1,2]
(1. 北京信息科技大学计算机学院,北京 100192;2. 未来区块链与隐私计算高精尖创新中心,北京 100191)

摘 要: 为了解决真实Web应用攻击数据数量小、差异性大和攻击载荷多样化导致大模型训练效果差的问题,提出一种基于联邦大模型的网络攻击检测方法(Intrusion Detection methods based on Federal Large Language Model,FL-LLMID). 首先,提出一种面向大模型微调的联邦学习网络,服务器对客户端本地大模型通过增量数据训练产生的参数,进行增量聚合的方式,提高联邦学习中大模型的参数聚合效率以及避免网络流量数据暴露的问题;其次,基于大模型对代码的理解能力,提出面向应用层数据的攻击检测模型(CodeBERT-LSTM),通过对应用层数据报文进行分析,使用CodeBERT模型对有效字段进行向量编码后,结合长短期记忆网络(Long Short-Term Memory,LSTM)进行分类,实现对Web应用高效的攻击检测任务;最后,实验结果表明,FL-LLMID方法在面向应用层数据的攻击检测任务中准确率达到99.63%,与传统联邦学习相比,增量式学习的效率提升了12个百分点.

关键词: 联邦学习;大模型;长短期记忆网络;CodeBERT;网络攻击检测;增量聚合
基金项目: 国家社会科学基金(No.21BTQ079);未来区块链与隐私计算高精尖中心基金(No.GJJ-24)
中图分类号: TP309 **文献标识码:** A **文章编号:** 0372-2112(2025)06-1792-13
电子学报URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20241098
第二十七届中国科协年会学术论文

Research on Network Attack Detection Method Based on Federated Large Model

KANG Hai-yan[1,2], ZHANG Yi-fan[1,2], WANG Nan-min[1,2]
(1. *Computer School*, *Beijing Information Science and Technology University*, *Beijing* 100192, *China*;
2. *Beijing Advanced Innovation Center for Future Blockchain and Privacy Computing*, *Beijing* 100191, *China*)

Abstract: To address the issues of a small quantity, large variability of real Web application attack data and diverse attack payloads that lead to poor training effects of large models, a network attack detection method based on federated large model (FL-LLMID) is proposed. Firstly, a federated learning network for fine-tuning large model is proposed. The server conducts incremental aggregation on the parameters generated by the client's local large model through incremental data training, which improves the parameter aggregation efficiency of large model in federated learning and avoids the problem of network traffic data exposure. Secondly, based on the large model ability to understand code, an attack detection model for application layer data (CodeBERT-LSTM) is proposed. By analyzing the application layer data packets, the CodeBERT model is used to perform vector encoding on the valid fields, and then combined with the long short-term memory network (LSTM) for classification to achieve the attack detection task of Web applications. Finally, the experimental results show that the accuracy of the FL-LLMID method in the attack detection task for application layer data reaches 99.63%. Compared with traditional federated learning, the efficiency of incremental learning is improved by 12 percentage points.

Key words: federated learning; large model; LSTM; CodeBERT; attack detection; incremental aggregation
Foundation Item(s): National Social Science Foundation of China (No.21BTQ079); Beijing Advanced Innovation Center for Future Blockchain and Privacy Computing Fund (No.GJJ-24)

收稿日期:2024-12-05;修回日期:2025-04-17;责任编辑:覃怀银

1 引言

随着互联网的发展,Web应用[1]呈现出多元化和高效化的趋势,从技术架构、用户体验到个性化部署等方面均有显著进步,因此使用量逐渐增加,Web应用安全的问题也随之日益严重. Web应用通常面临SQL(Structured Query Language)注入、跨站脚本攻击(Cross Site Script,XSS)、远程代码执行(Remote Code Execution,RCE)、文件上传和XML(eXtensible Markup Language)外部实体注入(XML External Entity,XXE)等类型的攻击. 传统的基于流量的攻击检测方法只针对流量数据的十六进制内容进行特征提取和分类,虽然取得了不错的效果,但是越来越多的变式攻击手法使上述方法不能得到正确的分类结果. 对此,本文从应用层对流量数据进行分析,提出一种新的检测方法.

如图1所示,传统基于机器学习和规则的检测方法,在对用户输入的"star"参数值进行检测时,由于"eval"字符串后是函数的嵌套,并没有出现参数传递的行为,所以,"eval"会被当作简单的字符串处理,不会检测出攻击行为.

```
GET /index.php?star=eval(next(apache_request_headers())); HTTP/1.1
Host: 127.0.0.1:81                                          ①
Upgrade-Insecure-Requests: 1
User-Agent: system('cat /flag'); ②
Accept: text/html,application/xhtml+xml,application/xml;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: zh-CN,zh;q=0.9
Connection: close
```

图1　应用层数据包文本

大语言模型[2]的发展为自然语言处理带来了新的研究价值,同样给弥补上述问题带来了新的思路. 代码可以看作是一种特殊的语言,同样具有语法. 大模型可以通过代码预训练学习,识别出函数、字符串和调用关系. 对于图1中的内容,大模型可以推理出"apache_request_headers"函数是获取了请求头,然后通过"next"函数获取了请求头中的下一个元素值,即"system('cat /flag')",然后传递给"eval"函数执行系统命令.

虽然大模型从新的角度给予了攻击检测的思路,但面临着缺乏真实数据的挑战. 如图1中的数据除了可以获得攻击方式外,还可以得到"Host"值所对应的网站的"index. php"路由下,存在一个代码执行的安全漏洞,这意味着暴露了网站的脆弱性. 因此,网站的管理者为了降低安全风险,不会对外开放真实的网络数据,这造成了网络安全领域缺乏真实训练数据的现状,给大模型的应用带来的新的挑战.

针对上述问题与挑战,本文进行了深入研究,主要贡献如下:

(1)为了解决网站攻击形式多样化、大模型训练缺乏真实数据以及数据隐私安全的问题,提出一种基于联邦大模型的网络攻击检测方法(Intrusion Detection methods based on Federal Large Language Model, FL-LLMID),该方法提取网站数据文本内容的特征,结合上下文信息进行分类,实现攻击检测. 此外,通过联邦学习的隐私保护特性,实现多方数据协同训练大模型.

(2)提出一种面向大模型微调的联邦学习网络,可以在多个本地节点上进行大模型训练和微调,然后将模型的增量参数上传到中央服务器进行异步加密聚合,生成新的全局模型参数. 通过这种方式,实现模型的增量式微调学习,提高了模型微调效率和客户端模型对网络攻击的检测准确率.

(3)提出一种面向应用层数据的攻击检测模型,该模型基于CodeBERT字符编码器和长短期记忆网络(Long Short-Term Memory,LSTM),利用CodeBERT双向学习的特性,生成针对代码语义的特征向量. 同时,为了捕捉数据包中上下文参数的关联信息,用LSTM网络对向量进行上下文特征提取,有效提升针对应用层数据检测的准确率.

(4)模拟真实网络环境,在KDD99数据集、UNSW-NB15数据集和CICIDS2017数据集上对所提FL-LLMID方法进行实验和分析,并与现有方法进行对比,验证了FL-LLMID方法的有效性、优越性和鲁棒性.

2 相关工作

针对网络攻击检测[3]的问题,众多人工智能算法应用于其中. 决策树(Decision Tree, DT)[4]因其复杂度低、易于理解,被广泛应用于入侵检测. 文献[5]提出了采用融合随机森林模型进行特征转换,梯度提升决策树模型进行分类的新模型RF-GBDT,该模型针对入侵检测数据不平衡的问题对传统的单一决策树模型进行了改进,但是该方法检测效率低并且训练时间长,不具有实时性. 朴素贝叶斯(Naive Bayes,NB)[6]可以更好地解决模型实时性的问题,文献[7]结合该模型,基于特征选择技术筛选出的重要特征,在ITD-UTM数据集上的实验证明对称不确定性(Symmetrical Uncertainty, SU)结合NB模型可以提高检测精度和速度,适应大流量网络环境. 文献[8]提出一种结合属性加值算法的改进贝叶斯模型,旨在简化数据分类复杂性,在KDD99数据集上的实验证明该模型分类精度显著提高,并且通过控制模型参数提高了分类效率. 上述基于机器学习的方法虽然取得了一定的效果,但严重依赖于人工对特征的提取工作.

为进一步挖掘样本特征,一些深度学习模型被应用在攻击检测问题中. 文献[9]提出一种基于循环神经网络[10](Recurrent Neural Network,RNN)的入侵检测系

统,通过RNN建模学习正常的网络行为识别出异常行为,从而实现对网络入侵的检测. 然而,RNN模型存在梯度消失和梯度爆炸问题,导致在处理长序列数据时效果不佳. 长短期记忆网络[11]和门控循环单元[12](Gated Recurrent Unit,GRU)是RNN的改进变体,通过引入门控机制,能够有效地解决梯度问题,更好地记忆长期信息. 文献[13]提出了基于LSTM的入侵检测系统,通过对一段时间内的网络流量数据进行建模分析. 实验结果表明,LSTM模型在检测持续性攻击和具有时间相关性的攻击行为时表现出色. GRU也在入侵检测中得到了应用,文献[14]提出了基于GRU的入侵检测系统,研究发现GRU在计算效率上相对LSTM有一定优势,同时在保持较好检测性能的前提下,能够更快地对新的入侵模式进行学习和适应. 但是,无论是基于GRU或是基于LSTM的方法,都会出现一定程度的信息丢失或难以完全捕捉到长序列中的所有重要信息. 为了解决这一问题,文献[15]提出结合Transformer[16]的双向GRU入侵检测方法,通过Transformer的注意力机制来捕捉数据的全局关系和重要特征,帮助BiGRU[17]更好地处理数据的上下文信息和长期关系.

针对网络数据的隐私保护问题,文献[18]提出了一种使用联邦学习的可靠的工业物联网异常检测策略. 该策略应用联邦学习技术来构建通用异常检测模型,训练本地深度强化学习算法,减少了隐私泄露的机会,实现了工业互联网数据的保护隐私. 文献[19]提出了一种安全高效的智能电网入侵检测方法,旨在保护本地数据并扩充数据量,通过安全协议防止攻击者窃听和推理. 文献[20]提出基于安全联邦蒸馏GAN的工业入侵检测方法(FL-SEResNet),该方法可以在免受隐私泄露的风险同时,生成数据来增强分类性能,在CPS数据集和AWID数据集上分别进行实验并取得了不错的效果. 文献[21]构造了一种面向异构环境的联邦学习框架(FedTP),该算法解决了节点模型在资源受限和数据非独立同分布时出现效率低、性能差的问题. 文献[22]结合联邦学习提出一种基于联邦学习和长短期记忆网络的智能入侵检测,该系统针对用户输入的复杂性和shell命令的上下文相关性,基于开源SEA数据集进行模型性能测试,结果表明,该方法能够在保证用户隐私的前提下学习用户服务器数据集的特征并且分类精度较高.

梳理上述研究后发现,该领域早期的文献[5,7,8]使用单一和优化后的机器学习模型,对网络环境中特定类型的攻击进行检测,填补了传统基于规则库匹配的攻击检测机制的空缺,为网络安全结合人工智能的研究奠定了基础. 文献[9]应用了深度学习模型对特征进行深度挖掘,但是带来了梯度消失和梯度爆炸问题,导致无法处理长序列数据. 文献[13~15]针对这一问题进行了改进,成功捕捉到数据的上下文信息,丰富了对Web应用攻击的检测与防护方法. 但由于网络环境不断地更新,攻击者的攻击方式千变万化,导致网络上公开的模型训练数据变得陈旧并且缺乏真实性. 文献[18,20,21]将联邦学习框架与深度学习模型相结合,突破了数据孤岛的瓶颈,通过多方的数据共同训练深度学习模型进一步提高了模型检测的准确率. 文献[19,21]在智能电网等特定的网络环境,使用安全的通信协议抵抗了联邦学习框架存在的一些隐私攻击,提供了一种网络数据隐私保护的方法.

虽然上述文献推动了网络攻击研究领域的发展,为该领域作出了巨大贡献. 但这些方法仍然存在训练数据质量低、数量小和检测模型对变式攻击不能正确识别的问题,因此本文提出基于联邦大模型的网络攻击检测方法,利用大语言模型对文本的理解能力,设计面向应用层数据的攻击检测模型,以提取网络数据特征,增强对变式网络攻击行为的识别能力. 利用联邦学习的隐私保护特性,设计面向大模型微调的联邦学习网络,实现多方协作,提高数据的数量和质量.

3 FL-LLMID方法设计与分析

针对客户端网站流量数据的隐私安全和Web漏洞攻击问题,提出一种基于联邦大模型的网络攻击检测方法(FL-LLMID),旨在通过面向大模型微调的联邦学习网络,利用联邦学习的隐私保护和多方协作特性,解决大模型训练过程中数据量小、效率低和网站脆弱点暴露的问题. 通过面向应用层数据的攻击检测模型,利用大语言模型对文本的理解能力和LSTM对文本序列的特征提取能力,检测和发现网络数据中的攻击行为.

FL-LLMID方法全局框架如图2所示,包括客户端和聚合服务器. 其中,客户端主要负责下载全局增量参数,训练面向应用层数据的攻击检测模型(CodeBERT-LSTM)来开展攻击检测工作,之后将训练所得的增量参数上传至聚合服务器. 聚合服务器主要对客户端上传的参数进行增量式聚合,并将其下发迭代,不断优化客户端攻击检测模型,构建面向大模型微调工作的联邦学习网络.

3.1 面向大模型微调工作的联邦学习网络

通常客户端网站流量数据会涉及到网站的安全性问题,所以大多公开用于训练模型的数据都经过了特殊处理,而这些处理过的数据有失真实性,导致大模型训练效果不佳. 针对这一问题本文提出了一种面向大模型微调工作的联邦学习网络,通过联邦学习“数据可用不可见”的特性,联合多客户端共同微调训练大模型. 由于大模型拥有庞大的参数量,使得全参数微调会消耗大量的计算资源和通信成本,因此,高效的参数微

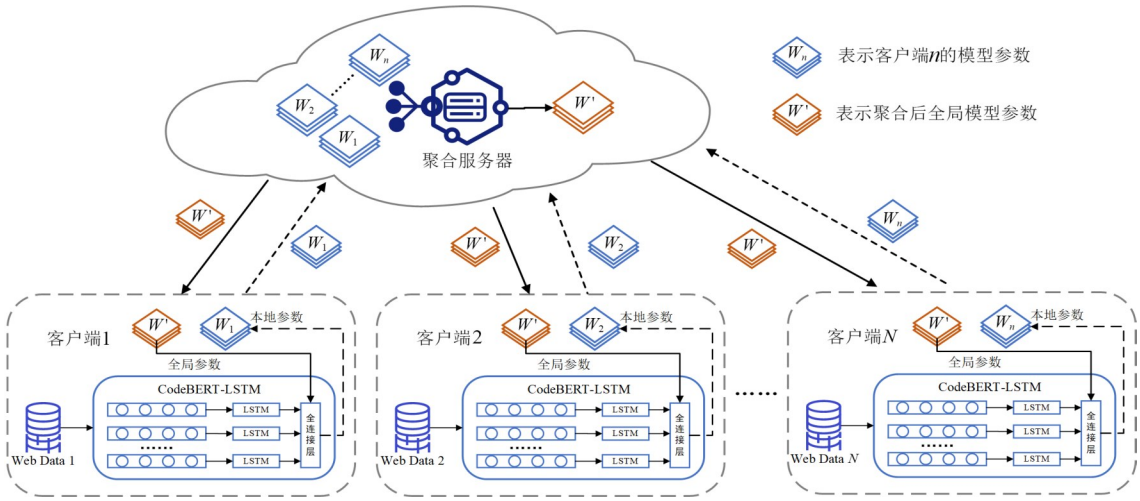


图2　FL-LLMID框架图

调方法成为大模型微调研究的热点内容. 针对这一问题提出一种基于异步联邦学习的增量式参数微调算法,通过部分参数微调解决效率低的问题,并设计一种加密聚合算法,以抵抗联邦学习框架中的推理攻击. 方法过程如图3所示.

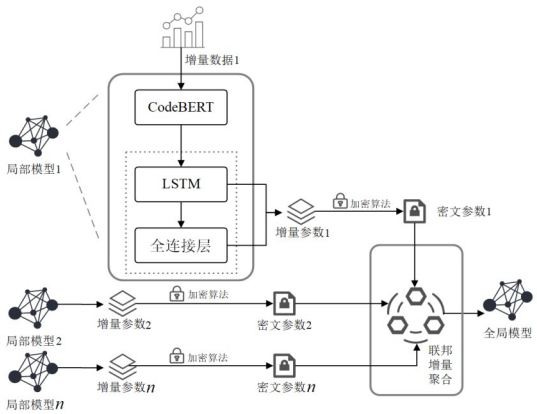


图3　增量式参数微调算法架构图

客户端通过增量数据训练局部模型后,在LSTM层和全连接层得到一组增量参数. 通过应用该模型于下游任务的过程可以发现,仅需对这一组增量参数进行微调,即可达到提升局部模型的效果,具体过程如下.

假设联邦学习网络中存在N个客户端,用n=1,2,3,…,N表示. 令θ_t表示在时间步t的全局模型参数,其中包括权重矩阵($\boldsymbol{W}_{\mathrm{if}}$、$\boldsymbol{W}_{\mathrm{hf}}$、$\boldsymbol{W}_{\mathrm{ii}}$、$\boldsymbol{W}_{\mathrm{hi}}$等用于输入门、隐藏门等的权重)和偏置向量($\boldsymbol{b}_{\mathrm{if}}$、$\boldsymbol{b}_{\mathrm{hf}}$、$\boldsymbol{b}_{\mathrm{ii}}$、$\boldsymbol{b}_{\mathrm{hi}}$等). 且初始时$t$=0,全局模型参数为$\theta_0$. 对于客户端$n$,本地数据集在时间步$t$表示为$D_{n,t}$,包含了之前的数据以及新到达的数据,本地LSTM模型参数为$\theta_{n,t}$. 对于LSTM单元,输入序列为$x=(x_1,x_2,\cdots,x_T)$,隐藏序列为$h=(h_1,h_2,\cdots,h_T)$,细胞状态序列为$c=(c_1,c_2,\cdots,c_T)$,在时间步t时,输入门i_t、遗忘门f_t、输出门o_t,如式(1)~式(3)所示.

$$i_t=\sigma\left(\boldsymbol{W}_{\mathrm{ii}}x_t+\boldsymbol{W}_{\mathrm{hi}}h_{t-1}+\boldsymbol{b}_{\mathrm{ii}}+\boldsymbol{b}_{\mathrm{hi}}\right) \tag{1}$$

$$f_t=\sigma\left(\boldsymbol{W}_{\mathrm{if}}x_t+\boldsymbol{W}_{\mathrm{hf}}h_{t-1}+\boldsymbol{b}_{\mathrm{if}}+\boldsymbol{b}_{\mathrm{hf}}\right) \tag{2}$$

$$o_t=\sigma\left(\boldsymbol{W}_{\mathrm{io}}x_t+\boldsymbol{W}_{\mathrm{ho}}h_{t-1}+\boldsymbol{b}_{\mathrm{io}}+\boldsymbol{b}_{\mathrm{ho}}\right) \tag{3}$$

其中,$\sigma(\cdot)$表示sigmoid激活函数,$\boldsymbol{W}$为权重矩阵,$\boldsymbol{b}$为偏置向量,细胞状态更新如式(4)所示:

$$c_t=f_tc_{t-1}+i_t\tanh\left(\boldsymbol{W}_{\mathrm{ic}}x_t+\boldsymbol{W}_{\mathrm{hc}}h_{t-1}+\boldsymbol{b}_{\mathrm{ic}}+\boldsymbol{b}_{\mathrm{hc}}\right) \tag{4}$$

输出的隐藏状态为$h_t=o_t\tanh(c_t)$.

客户端n根据本地数据$D_{n,t}$和当前本地模型参数$\theta_{n,t-1}$进行更新. 设损失函数为$L_n(\theta,D_{n,t})$,$m_{n,t}$和$v_{n,t}$分别为梯度一阶矩估计和二阶矩估计,在时间步t时,对于LSTM模型参数中索引为j的一个参数$\theta^j_{n,t-1}$,更新过程如式(5)~式(10)所示.

首先,计算梯度$g^j_{n,t}$:

$$g^j_{n,t}=\nabla L_n\left(\theta^j_{n,t-1};\,D_{n,t}\right) \tag{5}$$

更新一阶矩估计和二阶矩估计,其中β是优化器超参数:

$$m^j_{n,t}=\beta_1m^j_{n,t-1}+\left(1-\beta_1\right)g^j_{n,t} \tag{6}$$

$$v^j_{n,t}=\beta_2v^j_{n,t-1}+\left(1-\beta_2\right)\left(g^j_{n,t}\right)^2 \tag{7}$$

对一阶矩估计和二阶矩估计进行修正:

$$\hat{m}^j_{n,t}=\frac{m^j_{n,t}}{1-\beta_1^t} \tag{8}$$

$$\hat{v}^j_{n,t}=\frac{v^j_{n,t}}{1-\beta_2^t} \tag{9}$$

更新参数,其中η是学习率,ϵ为小常数,防止除数为0:

$$\theta^j_{n,t}=\theta^j_{n,t-1}-\frac{\eta}{\sqrt{\hat{v}^j_{n,t}}+\epsilon}\hat{m}^j_{n,t} \tag{10}$$

客户端n的权重为w_n,将本地更新后的模型参数更新量发送给聚合服务器,服务器对权重矩阵和偏置向量等参数进行聚合,如式(11)所示:

$$\boldsymbol{W}_{\mathrm{ii},t}=\sum_{n=1}^{N}w_n\boldsymbol{W}_{\mathrm{ii},n,t} \tag{11}$$

其中,$\boldsymbol{W}_{\mathrm{ii}}$表示权重矩阵,以此方法聚合其他参数,得到全局模型参数θ_t.

真实的HTTP数据可能包含敏感信息(如用户身份、登录凭证、交易数据等),网络攻击检测模型需要处理这些数据以识别攻击行为,因此,为了保护这些信息不会受到诚实但好奇的聚合服务器推理攻击,在参数上传聚合的过程中使用了Paillier同态加密,每个客户端将局部模型的增量参数加密上传到服务器,并且服务器仅对这些增量参数进行加密聚合,这很大程度上减小了计算资源和通信成本,提升了联邦微调的效率.其中,服务器的加密聚合算法如算法1所示,主要参数如表1所示.

算法1 服务器加密聚合算法

输入:初始化参与方列表$S_{parties}$=['$party_1$','$party_2$','$party_3$',…,'$party_n$']

输出:客户端解密全局模型参数$W_{decrypted}$

1. 参与方生成公钥K_{pub}和私钥K_{pir}
2. FOR $party_n$ IN $S_{parties}$
3. 参与方加密增量参数$\Delta_{data\ n}$ => $\Delta_{encrypted\ n}$
4. 生成参与方增量密文集合C_{args}=['$\Delta_{encrypted\ 1}$', '$\Delta_{encrypted\ 2}$', '$\Delta_{encrypted\ 3}$', …, '$\Delta_{encrypted\ n}$']
5. 初始化全局聚合结果 aggregated_result = [0] * len(encrypted_data[parties[0]])
6. FOR i IN C_{args}
7. 对密文数据进行同态加法聚合$W_{decrypted}$ += encrypted_data[party][i] + shared_encrypted_data[party][other_party][i]
8. 下发全局模型参数密文$W_{encrypted}$
9. 客户端解密全局模型参数$W_{decrypted}$ = [K_{pub}.decrypt(r) FOR r IN $W_{decrypted}$]
10. RETURN $W_{decrypted}$

表1 加密聚合算法主要参数表

符号	释义
$S_{parties}$	联邦学习网络的参与方列表
$party_n$	第n个参与方
$W_{decrypted}$	全局模型参数明文
$W_{enerypted}$	全局模型参数密文
K_{pub}/K_{pir}	参与方用于加密局部参数的公钥/私钥
Δ_{data}/$\Delta_{encrypted}$	增量参数明文/密文
C_{args}	服务器局部增量参数密文集合

针对Web应用场景中的攻击检测问题,对所述算法进行了系统性分析.该算法采用Paillier加密方案,基于部分同态加密机制,其密文数据膨胀率控制在原始数据的2倍以内,有效降低了存储与通信开销.同时,FL-LLMID方法采用了增量参数选择性加密策略,仅对模型更新部分的参数进行加密处理,从而在保证安全性的同时显著提升了计算效率.此外,基于同态加密的数学特性,服务器在密文空间内的计算结果与明文空间保持同态一致性,这使得加密过程对模型聚合精度的影响较小.

综合上述分析,尽管FL-LLMID方法在联邦学习过程中引入了额外的加密计算开销,但该代价对于保障Web应用数据的安全性具有必要性,特别是在应对SQL注入、XSS攻击等Web安全威胁时,这种安全性与效率的权衡具有重要的意义.

3.2 面向应用层数据的攻击检测模型

针对应用层数据包中存在众多需予以分析的字段特征这一问题,提出一种面向应用层数据的攻击检测模型(CodeBERT-LSTM).该模型旨在提取并剖析应用层数据包的复杂字段数据所蕴含的文本序列关联性特征,进而达成对应用层数据包的实时检测目标.

3.2.1 CodeBERT-LSTM模型结构

本文将Web应用所输入的HTTP数据选定为核心研究对象,鉴于HTTP数据包参数字段呈现出的上下文连续性特征,针对同一数据包的请求头与请求体展开深入分析具有重要意义.其中,请求头中的资源统一资源标识符(Uniform Resource Identifier,URI)与请求体的参数及参数值之间存在着显著的关联性,这种关联性能够作为判定数据包是否存在攻击行为的关键上下文依据.基于此,采用LSTM结构针对由CodeBERT模型所生成的文本向量实施特征提取操作,随后借助全连接层针对样本进行多分类处理,实现对Web应用的流量数据进行攻击检测的任务,有效提升对网络攻击的识别准确度.

CodeBERT-LSTM模型可划分为3个主要组成部分,其结构展示如图4所示.首先,数据预处理部分去除数据包内无显著语义价值的字段,并对数据格式实施标准化操作.以图4左侧所示数据包为例,其中"Connection: close"为通信协议所必需字段,该字段的变动对网站功能及安全性并无实质性影响,因此在数据预处理流程中予以删除,以精简数据规模并聚焦于关键信息.第2部分聚焦于文本处理流程,对经预处理后的文本执行分词操作,继而附加首端标签"<s>"与末端标签"</s>",从而构建成特定的字符序列.在此基础上,分别对字符序列进行词嵌入与位置嵌入,将字符序列精确映射为具有固定维度的向量表示形式,此向量能够有效捕捉文本的语义及结构信息,为后续特征提取奠定基础.第3部分核心在于利用LSTM结构针对文本向量执行深度特征提取任务,LSTM凭借其对序列数据的良好处理能力,能够有效挖掘文本向量中的时序及语义关联特征.随

后,借助全连接层与Softmax函数的联合运作,实现对文本的分类处理,最终获取文本的分类结果,以此达成对应用层数据包的精准分析与分类判定.

3.2.2 攻击检测算法

利用WireShark网络分析工具,依托上述经过联邦学习网络训练所获得的CodeBERT-LSTM模型,创新性地提出一种应用层数据攻击检测算法,其流程如图5所示.

对于客户端实时捕获的流量数据,借助WireShark工具进行解析,精准提取数据包中的字段信息,以文本的形式输入到CodeBERT-LSTM模型进行数据特征提取与深度分析,最终输出分类结果.具体分类结果参考OWASP(开放式Web应用程序安全项目)TOP10中的漏洞,共包含正常数据、SQL注入攻击、文件上传攻击等8类数据.为了让模型保持最优的性能,该算法在聚合服务器产生新的全局参数时,下载全局参数并对CodeBERT-LSTM模型进行更新迭代.算法的具体过程如算法2所示,相关参数见表2.

4 实验与分析

4.1 环境设置

4.1.1 拓扑仿真模拟

在模拟真实网络环境的研究过程中,采用Mininet(一种进程虚拟化网络仿真工具)来构建网络拓扑结构,其结构如图6所示.该拓扑图中呈现出3个客户端,运用3个路由器将这些客户端划分至3个局域网之中.其中,每个客户端主要由局部模型、Web服务器以及数据存储单元所构成.各个客户端负责收集与其自身服务器相关的流量数据,并且所有客户端均借助联邦学习服务器来实现模型的更新迭代过程.该环境为后续相关研究如基于联邦学习的模型训练、网络性能分析以及安全机制验证等提供了一个真实且可控的实验环境基础,有助于深入探究网络系统在复杂环境下的运行特性与规律.

4.1.2 实验环境与实验数据

实验环境为CPU:Intel(R) Xeon(R) Gold 5218,64核128线程,128 GB内存;GPU:Nvidia A100,40 GB显存;Ubuntu系统.所设计代码开发环境为PyCharm,采用Python实现.

实验数据集包括KDD99、UNSW-NB15、CICIDS2017、Maple-ID,其特征描述如下:

(1)KDD99数据集.从DARPA网络数据集文件创建.此数据集内含7周的网络流量,共490万条记录.攻击类型包括Denial of Server、Remote to User、User to Root和Probing.每个实例由3个类别的41个特征表示,分别是基本特征、流量特征和内容特征.其中,内容特征与数据部分的可疑行为有关,选取内容特征部分进行实验.

(2)UNSW-NB15数据集.由澳大利亚网络安全中心的靶场实验室创建.因其具有各种新颖的攻击方式,被广泛使用.攻击类型包括Fuzzer、Analysis、Backdoor、DoS、Exploits、Generic、Reconnaissance、Shellcode 和Wroms.该数据集中训练集、测试集分别包含82 332、175 341条记录.

(3)CICIDS2017数据集.包含良性和常见的攻击,包括源数据(PCAP)和网络流量分析结果,数据集内容基于HTTP、HTTPS、FTP、SSH和电子邮件协议的25个用户的抽象行为.

(4)HIKARI-2021数据集.该数据集由真实网络流量数据和模拟攻击流量组合而成,包含了SQL注入、XSS、文件上传等不同类型的Web应用攻击数据,具有真实性和高适用性的特点.

(5)Maple-ID数据集(2024).一个针对入侵检测系统(Intrusion Detection System,IDS)的网络流量数据集,模拟了真实网络环境,包含多种攻击类型和正常流量,具有多样性、真实性、结构化特征的特点,使其适用于机器学习和深度学习模型的开发和评估,并广泛应用于网络安全研究当中.

考虑数据集差异对实验的影响,分别使用KDD99数据集、UNSW-NB15数据集和CICIDS2017数据集按照20%、30%和50%的比例构成联合数据集进行实验,共90 252条数据,具体数据样本比例如表3所示.

因为本文重点研究Web应用所涉及的应用层攻击行为,参考OWASP TOP10中的漏洞,筛选数据类别及数目如表4所示.

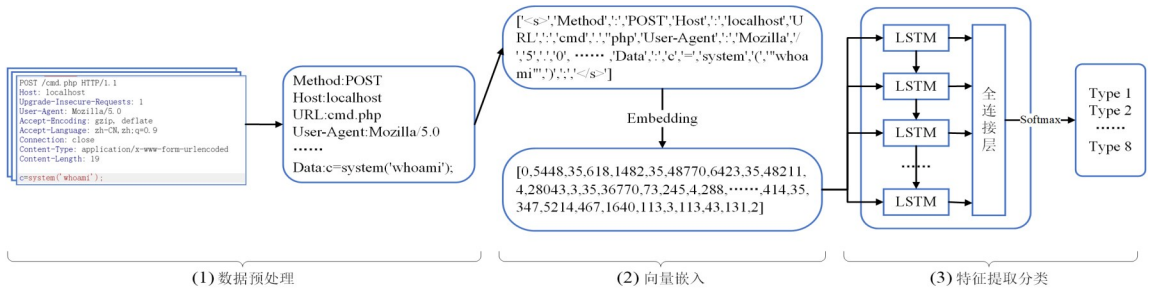


图4　CodeBERT-LSTM检测模型结构

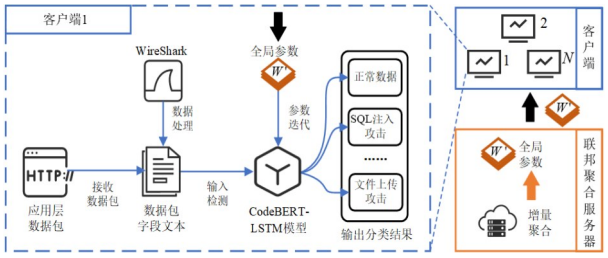


图5 攻击检测算法流程图

算法2 应用层攻击检测算法

输入:Wireshark解析得到的应用层数据包P_{data}和更新后的模型参数W'

输出:具有攻击行为的流量日志S_{attack}

1. 检查P_{data}请求协议,丢弃非http/https请求包
2. 提取请求参数item并初始化空文本信息P_{text}
3. 设置无效参数列表D_{args}
4. FOR item NOT IN D_{args}
5. 获取参数item对应的键值对K_{item},并加入文本信息P_{text}
6. IF W' IS NOT NULL
7. 检测模型Model对更新参数$W'(W_1+W_2+\cdots+W_n)$进行迭代
8. END IF
9. P_{text}输入Model进行分析,返回结果R_{text}
10. IF R_{text} IS NOT "valid"
11. 记录文本信息P_{text}到日志S_{attack}
12. 返回日志S_{attack}

表2 应用层攻击检测算法主要参数表

符号	释义
P_{data}	经过Wireshark解析得到的应用层数据包
W'	聚合服务器更新后的模型参数
S_{attack}	记录的流量日志
P_{text}	用于存放数据包有效字段的空文本信息
Model	CodeBERT-LSTM模型
$W_1+W_2+\cdots+W_n$	n个客户端上传的参数共同聚合
R_{text}	模型对数据包的分类结果
valid	没有攻击行为的正常数据

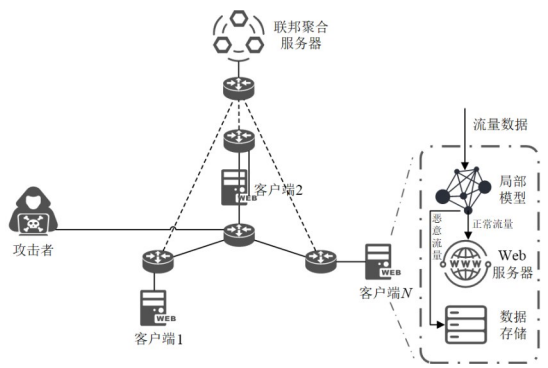


图6 仿真实验网络拓扑图

实验过程中,CodeBERT-LSTM模型训练参数如表5所示. 模型单次训练样本数(Batch_size)为128,学习率为5×10^{-5},共设置局部训练26轮和联邦学习全局训练5轮.

4.2 实验结果分析

4.2.1 性能实验

本节实验对所提FL-LLMID方法中的面向应用层数据的攻击检测模型(CodeBERT-LSTM)进行有效性验证. 模型训练迭代26轮次后结果如图7(a)和图7(b)所示. 分析实验结果可知:

(1)随着训练轮次的增加,Loss值逐渐减小并最终趋于稳定状态,证明模型性能表现逐渐稳定并保持在较高的水平.

(2)模型准确率从初始的73.30%以较快的收敛速度达到90%以上后,以较缓慢的速度提升,在第21轮次的迭代中达到最大值99.63%,并保持在99.00%以上. 其次,F1-score评价指标在完整训练过程中最大值达到99.14%后趋于稳定,证明了FL-LLMID方法在面向应用层数据的攻击检测任务中有优越的性能.

为了更好地评估模型对每个标签的分类准确性,对FL-LLMID方法在测试集样本上的攻击检测结果生成混淆矩阵,如图8所示. 矩阵中所有的数值均为概率值,其中,每行位于对角线位置的概率,代表着对应类型攻击检测的准确率,而其余概率则意味着错误分类的概率. 在横纵坐标中,SQLI、Path-Tr、File-Up和Normal分别对应表4所示的SQL注入攻击、目录遍历攻击、文件上传攻击以及正常数据这几种情况.

通过对图8的分析可以发现:仅有0.2%的攻击会被误判为正常数据. 在对攻击类型进行分类时,Path-Tr被误分类为其他攻击类型的概率相对较高. 这是因为在真实的网络攻击行为里,目录遍历行为常常在其他攻击的特定步骤中出现,这一特征同时存在于多项攻击类型的特征之中,其余类型的分类准确率都在99%以上,进一步证明了本文模型性能的优越性.

4.2.2 对比实验

本节实验将所提出的FL-LLMID方法与现有方法(VAE-CWGAN[23]、FL-GRU[24]、基于组内聚合的联邦学习[25]、MHFL-ID[25]、Fed-GA-CNN-IDS[26])进行对比试验,对比所用方法在应用层攻击流量检测分类任务的准确率和F1-score的结果,如表6所示.

分析表6可知:

(1)文献[23]的VAE-CWGAN方法用到卷积神经网络,聚焦于不同模型共同学习的研究,通过多模型协同检测网络攻击,具有一定的检测能力,但未充分考虑应用层数据的文本关联性特点,对于Web应用中请求路径和请

表3　数据集样本比例表

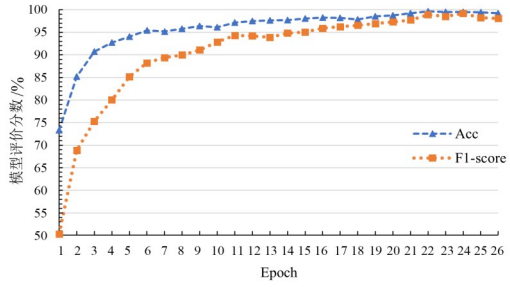
原始数据集	数据集良性样本数目	训练集攻击样本数目	测试集良性样本数目	测试集攻击样本数目
KDD99	7 844	5 694	2 346	2 167
UNSW-NB15	12 417	7 890	4 457	2 312
CICIDS2017	21 165	12 679	7 327	3 954

表4　数据类别及数目表

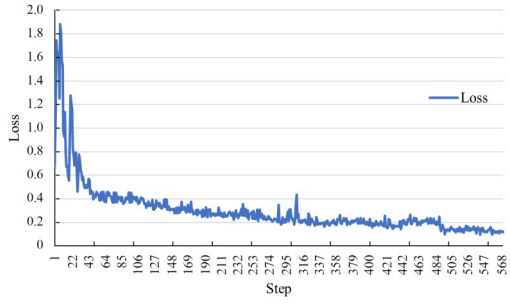
数据类别	数据数目
XSS(跨站脚本)攻击	6 446
RCE(远程命令执行)攻击	6 093
SQL注入攻击	7 269
目录遍历攻击	5 734
XXE(外部实体注入)攻击	8 645
SSRF(跨站请求伪造)攻击	7 870
文件上传攻击	3 070
正常数据	45 125

表5　训练参数表

参数	参数值
局部训练轮数	26
全局训练轮数	5
学习率	5×10^{-5}
Batch_size	128



(*a*) 模型准确率(Acc)和F1-score



(*b*) 模型损失值(Loss)

图7　FL-LLMID模型评估指标

求参数间的紧密联系挖掘不足,导致检测精度还有提升空间. 文献[24]的FL-GRU方法采用联邦学习框架结合GRU,一定程度上解决了梯度消失和梯度爆炸问题,在处理具有时间相关性的攻击行为检测时具有一定优势. 同时,联邦学习框架使其在一定程度上能够利用多方数据

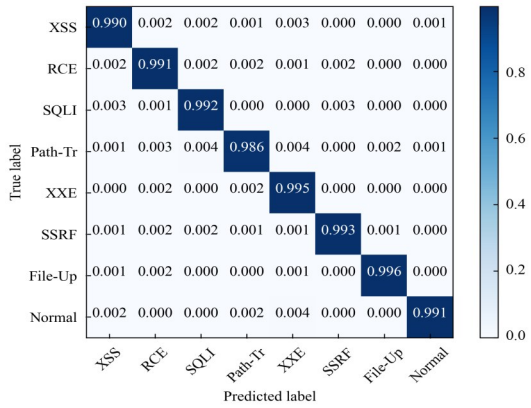


图8　多分类任务中FL-LLMID的混淆矩阵

表6　不同方法在入侵检测任务中的评估指标　　单位:%

方法模型	Acc	Rec	Prec	F1-score
VAE-CWGAN	92.36	93.83	92.16	93.41
FL-GRU	90.76	90.29	89.03	90.17
组内聚合&联邦学习	83.05	83.18	81.32	82.93
MHFL-ID	82.74	83.61	80.77	82.95
Fed-GA-CNN-IDS	91.05	93.20	89.82	92.18
FL-LLMID(本文)	99.63	99.98	97.84	99.14

进行训练. 但是GRU在处理长序列数据时仍会出现信息丢失问题,难以完全捕捉应用层数据包中的所有重要信息,导致对一些攻击行为的检测不准确. 文献[15]中方法仅依赖召回率和精准率作为双目标函数来选取组长,指标较为单一. 实际的入侵检测场景复杂多样,其他指标如误报率、漏报率等同样关键,忽略这些会使模型优化方向不全面. 并且这两个目标相互矛盾,难以同时最优,限制了模型整体性能的提升. 文献[26]的Fed-GA-CNN-IDS方法中,遗传算法的交叉概率、变异概率等参数的设置对结果影响较大,缺乏有效的自动调整机制,在不同Web应用下无法取得最佳效果.

(2)应用层数据解析后的文本内容,整体具有相关联系,特别是在请求路径和请求参数两部分会有较大的关联,本文方法针对上述特点,利用CodeBERT双向学习特性生成针对代码语义的特征向量,能够深入理解应用层数据中的代码含义. 结合LSTM网络对向量进行上下文特征提取,充分捕捉数据包中上下文参数的关联信息,因此在准确率和F1-score评价指标上取得了最优的效果.

4.2.3 鲁棒性实验

本节实验将所提FL-LLMID方法的客户端本地模型进行5轮全局训练后应用在各参与方，分别使用Maple-ID［https://maple.nefu.edu.cn/dataset］数据集、HIKARI-2021［https://zenodo.org/records/5199540］数据集和NSL-KDD数据集对参与方1、参与方2和参与方3进行攻击检测实验，以模拟评估模型在不同场景下的表现，并与全量学习方法进行对比.

图9所示为模型在不同参与方的局部数据集上的攻击检测准确率，对实验结果进行如下分析：

（1）对比4.2.1节微调模型所使用的数据集内容，本节实验所使用的数据集在数据的结构均为应用层HTTP协议，因此在数据的结构上没有产生大的变化. 其次，数据内容的差异主要产生于不同的Web应用，比如URL（统一资源定位符）、参数传递的参数名和参数值等，而攻击载荷当中的内容（例如SQL语句、命令执行函数）仍然存在于参数值当中. 对于这些内容，无论是上下文位置的改变，或是Web应用的变化，CodeBERT-LSTM模型都能够定位并且理解代码的语义，从而检测出是一条恶意的流量数据. 这验证了本文方法中的模型能够检测到新的攻击数据.

（2）全量学习方法在不同参与方上的准确率存在相对较大的波动，并且准确率的峰值和均值分别低于本文的增量学习方法5.15个百分点和5.40个百分点，结果表明，本文所提增量学习的方法对于不同局部数据集上的攻击检测任务有较高的鲁棒性.

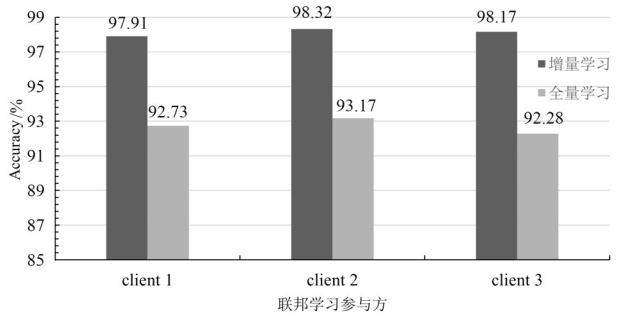


图9 不同参与方攻击检测鲁棒性实验

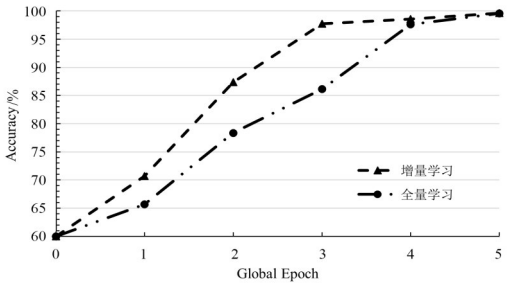
4.2.4 模型聚合对比实验

除了关注攻击检测任务的准确率问题，本节还对联邦学习框架的学习效率进行实验分析，在大模型应用问题中，模型通常具有较多的训练参数和较高的计算复杂度，因此相比深度学习模型将会消耗更多的时间. 本节对模型不同的学习方法进行时间和准确率上的探究，实验结果如表7和图10所示.

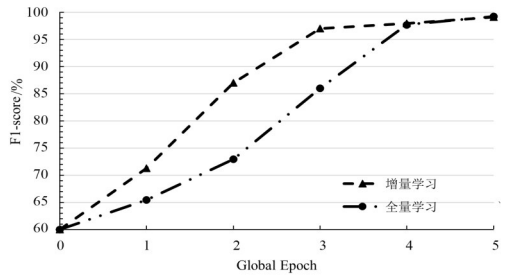
图10中横轴表示联邦学习全局训练轮次，纵轴分别表示准确率（Acc）和F1-score评价指标数值，分别展示了增量学习和全量学习的模型准确率和F1-score数值的对比情况. 其中，应用了增量学习方法的模型训练在收敛速度均优于全量学习，并且在最终的性能数值也占有一定优势，能够超过应用了全量学习的模型训练方式，这表明本文增量学习的模型训练方法在这个实验场景下是有效的.

表7 数据集样本比例 单位：%

学习方法	Acc	Rec	Prec	F1-score
全量学习	99.58	99.71	97.32	99.23
增量学习(本文)	99.63	99.98	97.84	99.14



(*a*) 以准确率为评估指标



(*b*) 以F1-score为评估指标

图10 增量学习方法有效性实验

为了证明本文增量学习方法的可行性，对比了2种学习方式训练所消耗的时间，实验结果如图11所示.

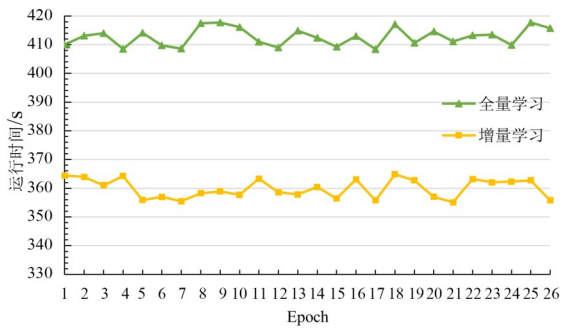


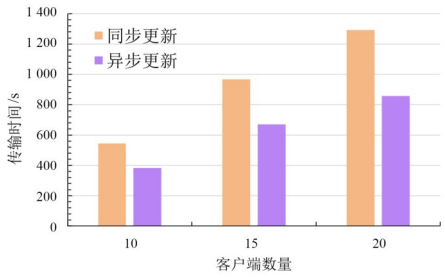
图11 不同学习方法下的时间对比

图11中，横轴表示参与方本地模型训练轮次，纵轴表示每一轮次训练所消耗的时间. 分析可得，全量学习的运行时间波动较大，总体上在400~420 s之间，增量学习的运行时间相对稳定，在350~370 s之间，平均每

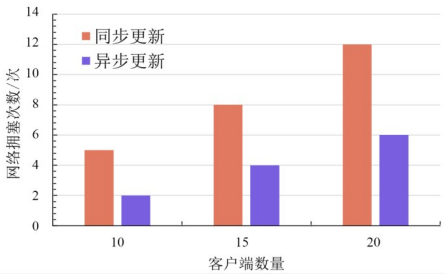
一轮次训练中比全量学习方法减少12%的运行时间，证明了本文增量学习方法在实验场景下的可行性.

4.2.5　通信开销实验

为了验证FL-LLMID方法的异步联邦学习能够在真实的大规模网络环境中具有更高的通信效率，本节实验设置了10、15、20个客户端和参与联邦学习，并设定网络带宽为100 Mbit/s，对比同步更新和异步更新策略下的通信开销. 异步更新策略为客户端在不同时间点上传增量参数，服务器在收到50%客户端数量的参数后进行聚合，并及时将更新后的全局模型参数下发给已上传参数的客户端. 实验结果如图12所示.



(a) 传输时间对比



(b) 网络拥塞对比

图12　不同更新策略下的传输时间对比

根据实验结果可知：

(1)随着客户端数量增加，同步更新和异步更新的总传输时间都在增长，但异步更新策略下的总传输时间始终低于同步更新. 在10个客户端时，异步更新比同步更新节省163 s；15个客户端时，节省296 s；20个客户端时，节省435 s. 这表明异步更新策略能有效分散数据传输压力，减少等待时间，提高通信效率.

(2)同步更新策略下，网络拥塞次数随着客户端数量增加而显著增多，因为大量客户端同时上传数据易造成带宽拥堵. 而异步更新策略下，拥塞次数明显减少，仅为同步更新的40%~50%左右，说明异步更新能更好地利用网络带宽，避免网络拥塞.

通过上述实验结果可知，在不同数量客户端参与的大规模网络环境中，FL-LLMID方法中的异步联邦学习在通信效率方面优于同步更新策略，同时，异步更新策略能够有效减少总传输时间和网络拥塞次数，一定程度上缓解联邦学习网络在真实环境中的通信代价.

4.2.6　模型消融实验

为了验证FL-LLMID方法中CodeBERT以及LSTM两个模型相比于其他的文本处理模型、序列处理模型具有更好的检测效果，在4.2.3节实验的数据集上分别使用代码预训练模型Code Llama[27]和BiGRU[28]模型进行替换实验，实验结果如表8所示，具体实验设置如下：

(1)CodeBERT-BiGRU. 将LSTM替换为BiGRU与CodeBERT进行组合，目的是验证本文方法中针对Web应用的流量数据使用LSTM进行序列处理具有更好的效果.

(2)Code Llama-LSTM. 用Code Llama模型替换CodeBERT与LSTM进行组合，目的是验证CodeBERT在处理HTTP请求的文本内容这一关键步骤优于同类型模型.

(3)Code Llama-BiGRU. 将Code Llama与BiGRU进行组合，验证本文方法所使用的文本处理模型和序列处理模型的组合在检测应用层数据任务中具有更好的优势.

表8　消融实验结果　　单位:%

模型组合	Acc	Rec	Prec	F1-score
CodeBERT-BiGRU	96.84	95.88	97.92	97.14
Code Llama-LSTM	98.12	97.34	98.01	97.85
Code Llama-BiGRU	95.36	96.29	95.81	95.93
CodeBERT-LSTM(本文)	99.63	99.98	97.84	99.14

对表8的实验结果进行分析，对比2、3组的实验，可以发现LSTM在对文本处理模型得到的特征向量进行信息捕捉时，相比BiGRU能够取得较好的效果. 此外，对比2、4组实验，采用CodeBERT的组合占有较高的优势. 针对这一结果，分析如下：

(1)BiGRU尽管通过双向机制增强了对全局上下文的捕获能力，但其相对简化的门机制限制了对复杂长距离依赖关系的建模能力. 在Web应用攻击检测任务中，HTTP数据包往往包含跨字段或跨时间步的潜在攻击特征(例如SQL注入的攻击载荷)，这些复杂依赖超出了BiGRU主要适用于局部上下文捕获的范围. 此外，双向传播在处理长序列时可能导致梯度衰减问题，进一步削弱其对长序列特征的建模效果，使其在需要高精度的复杂场景中表现不如LSTM优越.

(2)Code Llama基于Decoder-only架构，采用因果语言建模，主要面向生成任务. 这种单向上下文处理方式虽然适合生成序列数据，但在分析HTTP数据包时，难以充分捕获其双向关联的结构化特征(如请求头与正文、URL参数与方法等的语义依赖关系). 此外，Code Llama的大规模参数增加了微调和推理的资源需求，不

适合高效处理实际HTTP流量的大规模分析. 相比之下,HTTP数据包具有强结构性和多字段间复杂关联的特点,而CodeBERT的Encoder-only架构通过双向自注意力机制,能够同时建模请求与响应中的全局语义关系,更加契合入侵检测任务中对异常模式和上下文依赖的全面理解需求,且在资源受限场景中更具优势.

5 结束语

本研究围绕通过应用层数据特征对Web流量进行攻击检测问题,鉴于传统深度学习模型在应对多变攻击载荷时存在的局限性,以及大模型训练过程中所面临的数据量匮乏与数据涉及网络敏感脆弱信息等难题,创新性地提出了一种基于联邦大模型的网络攻击检测方法(FL-LLMID). 该方法包括面向大模型微调任务的联邦学习网络以及面向应用层数据的攻击检测模型(CodeBERT-LSTM)2个创新部分,并通过实验充分证实了此方法在多用户协同训练模型场景中的有效性,且攻击检测模型相较于其他同类方法在准确率与F1-score这2项关键评价指标上均展现出显著的优越性与提升态势.

展望未来研究方向,将持续致力于对所提出方法的深度完善与优化改进. 其一,着力探索在联邦大模型微调任务运行进程中所潜藏的模型安全与数据安全隐患问题,结合差分隐私[29]技术构建更为稳固可靠的安全防护机制,确保模型训练与应用过程的安全性与稳定性. 其二,深入挖掘如何充分发挥大模型对代码文本的深度理解能力,以此拓展识别更多新型网络攻击行为的有效途径与策略,增强网络攻击检测的全面性与精准度. 其三,紧密贴合实际应用场景需求,全面探索该方法在车联网[30]、工业互联网[31]以及通信网络等多领域中的具体应用模式与适配性优化方案,推动该方法在不同网络环境下的广泛应用.

参考文献

[1] 何海涛, 许可, 杨帅林, 等. 基于博弈的Web应用程序中访问控制漏洞检测方法[J]. 通信学报, 2024, 45(6): 117-130.
HE H T, XU K, YANG S L, et al. Game-based detection method of broken access control vulnerabilities in Web application[J]. Journal on Communications, 2024, 45(6): 117-130. (in Chinese)

[2] 周昆, 朱余韬, 陈志朋, 等. YuLan-Chat: 基于多阶段课程学习的大语言模型[J]. 计算机学报, 2025, 48(1): 1-18.
ZHOU K, ZHU Y T, CHEN Z P, et al. YuLan-chat: A large language model based on multi-stage curriculum learning[J]. Chinese Journal of Computers, 2025, 48(1): 1-18. (in Chinese)

[3] 仇晶, 陈荣融, 朱浩瑾, 等. 基于溯源图的网络攻击调查研究综述[J]. 电子学报, 2024, 52(7): 2529-2556.
QIU J, CHEN R R, ZHU H J, et al. A survey of network attack investigation based on provenance graph[J]. Acta Electronica Sinica, 2024, 52(7): 2529-2556. (in Chinese)

[4] 郭艳卿, 王鑫磊, 付海燕, 等. 面向隐私安全的联邦决策树算法[J]. 计算机学报, 2021, 44(10): 2090-2103.
GUO Y Q, WANG X L, FU H Y, et al. Federated decision tree algorithm for privacy security[J]. Chinese Journal of Computers, 2021, 44(10): 2090-2103. (in Chinese)

[5] MADHAVI M,NETHRAVATHI D R.Gradient boosted decision tree(GBDT) and grey wolf optimization(GWO) based intrusion detec-tion model[J]. Journal of Theoretical and Applied Information Technology, 2022, 100(16): 4937-4951.

[6] 李梓童, 孟小峰, 王雷霞, 等. 机器遗忘综述[J]. 软件学报, 2025,36(4):1637-1664.
LI Z T, MENG X F, WANG L X, et al. Survey on Machine Unlearning[J]. Software Journal, 2025, 36(4): 1637-1664. (in Chinese)

[7] STIAWAN D, HERYANTO A, BARDADI A, et al. An approach for optimizing ensemble intrusion detection systems[J]. IEEE Access, 2020, 9: 6930-6947.

[8] 王辉, 陈泓予, 刘淑芬. 基于改进朴素贝叶斯算法的入侵检测系统[J]. 计算机科学, 2014, 41(4): 111-115, 119.
WANG H, CHEN H Y, LIU S F. Intrusion detection system based on improved naive Bayesian algorithm[J]. Computer Science, 2014, 41(4): 111-115, 119. (in Chinese)

[9] FU Z Y. Computer network intrusion anomaly detection with recurrent neural network[J]. Mobile Information Systems, 2022, 2022(1): 6576023.

[10] 汪定, 邹云开, 陶义, 等. 基于循环神经网络和生成式对抗网络的口令猜测模型研究[J]. 计算机学报, 2021, 44(8): 1519-1534.
WANG D, ZOU Y K, TAO Y, et al. Password guessing based on recurrent neural networks and generative adversarial networks[J]. Chinese Journal of Computers, 2021, 44(8): 1519-1534. (in Chinese)

[11] 尹梓诺, 马海龙, 胡涛. 基于联合注意力机制和一维卷积神经网络-双向长短期记忆网络模型的流量异常检测方法[J]. 电子与信息学报, 2023, 45(10): 3719-3728.
YIN Z N, MA H L, HU T. A traffic anomaly detection method based on the joint model of attention mechanism and one-dimensional convolutional neural network-bidirectional long short term memory[J]. Journal of Elec-

tronics & Information Technology, 2023, 45(10): 3719-3728. (in Chinese)

[12] 袁文浩, 胡少东, 时云龙, 等. 一种用于语音增强的卷积门控循环网络[J]. 电子学报, 2020, 48(7): 1276-1283.
YUAN W H, HU S D, SHI Y L, et al. A convolutional gated recurrent network for speech enhancement[J]. Acta Electronica Sinica, 2020, 48(7): 1276-1283. (in Chinese)

[13] YAO R Z, WANG N, LIU Z H, et al. Intrusion detection system in the advanced metering infrastructure: A cross-layer feature-fusion CNN-LSTM-based approach[J]. Sensors, 2021, 21(2): 21020626.

[14] XU C Y, SHEN J Z, DU X, et al. An intrusion detection system using a deep neural network with gated recurrent units[J]. IEEE Access, 2018, 6: 48697-48707.

[15] LU W J, SHI C, FU H, et al. A power transformer fault diagnosis method based on improved sand cat swarm optimization algorithm and bidirectional gated recurrent unit[J]. Electronics, 2023, 12(3): 12030672.

[16] XU L T, HU C H, HU Y, et al. UPT-Flow: Multi-scale transformer-guided normalizing flow for low-light image enhancement[J]. Pattern Recognition, 2025, 158: 111076.

[17] HE W, MA H Y, GUO R, et al. Enhancing the state-of-charge estimation of lithium-ion batteries using a CNN-BiGRU and AUKF fusion model[J]. Computers and Electrical Engineering, 2024, 120: 109729.

[18] WANG X D, GARG S, LIN H, et al. Toward accurate anomaly detection in industrial internet of things using hierarchical federated learning[J]. IEEE Internet of Things Journal, 2022, 9(10): 7110-7119.

[19] LIU C J, SHI R. A smart grid intrusion detection model for secure and efficient federated learning[J]. Netinfo Security, 2023, 23(4): 90-101.

[20] 梁俊威, 杨耿, 马懋德, 等. 基于安全联邦蒸馏GAN的工业CPS协作入侵检测系统[J]. 通信学报, 2023, 44(12): 230-244.
LIANG J W, YANG G, MA M D, et al. Secure federated distillation GAN for CIDS in industrial CPS[J]. Journal on Communications, 2023, 44(12): 230-244. (in Chinese)

[21] 刘静, 慕泽林, 赖英旭. 面向异构环境的物联网入侵检测方法[J]. 通信学报, 2024, 45(4): 114-127.
LIU J, MU Z L, LAI Y X. Intrusion detection method for IoT in heterogeneous environment[J]. Journal on Communications, 2024, 45(4): 114-127. (in Chinese)

[22] ZHAO R J, YIN Y, SHI Y, et al. Intelligent intrusion detection based on federated learning aided long short-term memory[J]. Physical Communication, 2020, 42: 101157.

[23] 刘涛涛, 付钰, 王坤, 等. 基于VAE-CWGAN和特征统计重要性融合的网络入侵检测方法[J]. 通信学报, 2024, 45(2): 54-67.
LIU T T, FU Y, WANG K, et al. Network intrusion detection method based on VAE-CWGAN and fusion of statistical importance of feature[J]. Journal on Communications, 2024, 45(2): 54-67. (in Chinese)

[24] TANG Z Y, HU H Y, XU C H. A federated learning method for network intrusion detection[J]. Concurrency and Computation: Practice and Experience, 2022, 34(10): e6812.

[25] 高迢康, 靳晓宁, 赖英旭. 模型异构的联邦学习入侵检测[J]. 北京工业大学学报, 2024, 50(5): 543-557.
GAO T K, JIN X N, LAI Y X. Model heterogeneous federated learning for intrusion detection[J]. Journal of Beijing University of Technology, 2024, 50(5): 543-557. (in Chinese)

[26] SHAO J M, ZENG G Q, LU K D, et al. Automated federated learning for intrusion detection of industrial control systems based on evolutionary neural architecture search[J]. Computers & Security, 2024, 143: 103910.

[27] ROZIÈRE B, GEHRING J, GLOECKLE F, et al. Code llama: Open foundation models for code[EB/OL]. (2024-11-31)[2024-12-05]. https://arxiv.org/abs/2308.12950v3.

[28] SHE D M, JIA M P. A BiGRU method for remaining useful life prediction of machinery[J]. Measurement, 2021, 167: 108277.

[29] 康海燕, 王骁识. 基于数据特征相关性和自适应差分隐私的深度学习方法研究[J]. 电子学报, 2024, 52(6): 1963-1976.
KANG H Y, WANG X S. Research on the deep learning method based on data feature relevance and adaptive differential privacy[J]. Acta Electronica Sinica, 2024, 52(6): 1963-1976. (in Chinese)

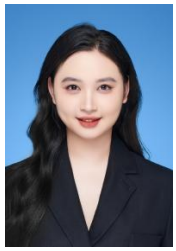
[30] 徐思雅, 郭佳惠. 基于双层联邦学习的高动态车联网业务边缘协作计算机制[J]. 电子学报, 2024, 52(7): 2228-2241.
XU S Y, GUO J H. Dual-layer federated learning based edge collaborative computing mechanism for high dynamic Internet of vehicle businesses[J]. Acta Electronica Sinica, 2024, 52(7): 2228-2241. (in Chinese)

[31] 龙墨澜, 康海燕. 基于代码可视化的工业互联网恶意代码检测方法[J]. 计算机集成制造系统, 2025, 31(2): 567-578.
LONG M L, KANG H Y. Detection method of industrial Internet malicious code based on code visualization[J]. Computer Integrated Manufacturing Systems, 2025, 31(2): 567-578. (in Chinese)

作者简介

康海燕 男,1971年生,河北石家庄人. 博士,北京信息科技大学计算机学院教授. 主要研究方向为网络安全与隐私计算等.
E-mail: kanghaiyan@126.com

张义钒 男,2000年生,内蒙古乌海人. 北京信息科技大学网络空间安全专业硕士研究生. 主要研究方向为网络安全与隐私保护等.
E-mail: zhangyf@126.com



王楠敏 女,2001年生,湖南株洲人. 北京信息科技大学电子信息专业硕士研究生. 主要研究方向为网络安全与隐私保护等.
E-mail: bellawang1105@126.com