计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
2025,34(7):96−106 [doi: 10.15888/j.cnki.csa.009858] [CSTR: 32024.14.csa.009858]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

应用于 Web 日志的持续学习异常检测系统①

鄢 薇[1], 李 畅[2], 田 征[1], 陈 凯[2], 李菁菁[2], 赵 静[2]

[1](国家气象信息中心, 北京 100081)
[2](中国科学院 计算机网络信息中心, 北京 100094)
通信作者: 赵 静, E-mail: jingzhao@cnic.cn

摘 要: 随着网络环境与攻击手段的变化, 大部分网络攻击检测的方法应用在真实场景中会随着时间的推移逐渐丧失高性能, 导致检测结果不稳定. 因此本文针对变化的真实网络攻击设计了一种基于极值理论的持续学习异常检测系统 E-TCEVT. 该系统的构建通过引入一种结合词级和子词级的混合语言模型, 用于从 Web 日志中有效提取特征. 在检测阶段, 采用基于极值理论和集成学习的思路, 通过集成多个基于不同时间点训练的模型防止模型微调时的灾难性遗忘, 实现模型对新旧样本的适应性和性能维持. 在开源数据集和真实数据集上的实验表明, 与单模型微调更新相比, 本文提出的方法具有更高的 $F1$ 分数; 与传统的非更新的方法相比, 本文方法在召回率和 $F1$ 分数上都表现更好.

关键词: 入侵检测; 向量化特征; 持续学习; 网络安全

引用格式: 鄢薇,李畅,田征,陈凯,李菁菁,赵静.应用于 Web 日志的持续学习异常检测系统.计算机系统应用,2025,34(7):96–106. http://www.c-s-a.org.cn/1003-3254/9858.html

Continual learning Anomaly Detection System Applied to Web Logs

FENG Wei[1], LI Chang[2], TIAN Zheng[1], CHEN Kai[2], LI Jing-Jing[2], ZHAO Jing[2]

[1](National Meteorological Information Center, Beijing 100081, China)
[2](Computer Network Information Center, Chinese Academy of Sciences, Beijing 100094, China)

Abstract: The performance of most Web attack detection systems degrades in real-world scenarios over time due to changes in the network environment and the evolution of attack techniques, resulting in unstable detection results. Therefore, this study designs an E-TCEVT anomaly detection system based on extreme value theory for dynamic real-world network attacks. This system incorporates a hybrid language model that combines word-level and subword-level elements for effective feature extraction from Web logs. In the detection phase, an approach based on extreme value theory and ensemble learning is employed. By integrating multiple models trained at different time points the proposed method prevents catastrophic forgetting during model fine-tuning, thereby maintaining adaptability and performance across both new and old samples. Experiments on open-source and real-world datasets demonstrate that the proposed method achieves higher $F1$ scores compared to single-model fine-tuning updates; moreover, compared to traditional non-updating methods, the proposed method shows better performance in both recall and $F1$ scores.

Key words: intrusion detection; vectorized feature; continual learning; network security

1 引言

互联网已经成为人们日常生活中不可或缺的一部分. 然而, 互联网也面临着越来越多的网络安全威胁, 这些威胁不仅会破坏互联网中各种资源的完整性和可

① 收稿时间: 2024-10-29; 修改时间: 2024-11-19; 采用时间: 2024-12-17; csa 在线出版时间: 2025-05-27
CNKI 网络首发时间: 2025-05-28

用性, 还会危及用户的隐私和数据安全. 由于 Web 应用程序广泛应用于各种领域, 如电子商务、金融、医疗等, 因此其安全性尤为重要. 这些应用程序通常包含大量用户数据和敏感信息, 如个人身份信息、财务记录和健康数据等, 一旦遭受攻击, 可能会导致重大的数据泄露和经济损失, 甚至威胁人们的生命安全. 业界现在已采用若干防御机制, 如 Web 应用程序防火墙 (Web application firewall, WAF)、入侵检测系统 (intrusion detection system, IDS) 等, 来确保 Web 应用程序的安全性. 为进一步提高 Web 应用程序安全防护水平, 研究人员研究了各种 Web 攻击检测方法. 在传统的 Web 攻击检测方法中, 主要采用特征提取和机器学习技术, 如支持向量机 (SVM)、决策树和随机森林等, 来检测 Web 攻击. 这些方法通常需要大量的人工特征提取和大规模的数据集来训练和测试模型, 并且模型在一定程度上存在性能不均衡的问题. 因此, 先前研究[1]针对该局限性, 引入集成学习以综合提高检测性能, 提出了 ELSV (ensemble learning classification with semantic vectorization) 系统来检测网站访问日志中的异常, 并在 CSIC 2010[2]数据集上验证了该系统的较高综合性能.

我们将 ELSV 应用到中国气象局真实环境中以检测网站访问日志数据中的异常. 一开始它表现出了良好的检测性能, 但随着时间的推移, 检测系统的性能随着时间出现了较大的起伏, 导致这一问题的原因可能是出现了新的日志. 并且在特征提取过程中出现越来越多的词汇外词, 使得 ELSV 中的词汇表和停用词表难以描述新的日志. Web 访问请求数据的不断变化, 模型学习到的数据分布难以适应新的日志样本. 然而目前已有的研究通常基于数据的平稳假设, 用训练数据的分布特征描述未来的样本, 但是这种假设无法适用于现在网络的 Web 访问日志的异常检测场景. 在现在网络环境中, 为了迎合新的用户需求, Web 服务本身会频繁发生业务调整和功能升级等; 一方面, 随着攻击者技术水平的提高, 为了绕过防御机制不断变换攻击方式, 甚至隐藏在正常请求中, 这种隐蔽性攻击也会导致日志发生改变. 另一方面, 在 Web 日志中, 请求中会出现新的资源地址以及新的请求参数等, 检测模型训练时所学习到的数据分布不再适用, 从而导致概念漂移问题[3,4].

Web 访问日志是文本数据, 其异常特征难以从统计方法中捕捉. Web 检测方法同时也面临概念漂移导致的检测性能降低的问题. 为了探究概念漂移对异常检测的影响, 我们选取了 4 种基于机器学习的 Web 异常检测方法[5–7], 并在开源数据集和中国气象局真实的网络数据集上进行测试, 发现非更新模型类的方法在面对概念漂移的数据时会发生不同程度的性能下降. 本文设计了一种基于持续学习的 Web 异常检测系统, 面对存在概念漂移的日志数据提出混合语言模型以实现从可能存在词汇外词的漂移样本中提取特征. 再基于极值理论和持续学习, 自动地筛选出偏离已知样本分布的样本, 利用这些被拒绝的样本训练新的子模型, 添加到异常检测系统中. 保持检测系统的性能稳定性, 从而主动地适应漂移数据, 本文的主要贡献如下.

1) 基于开源数据集和真实数据集对先进的 Web 异常检测方法进行评估, 分析了概念漂移问题对异常检测器的性能影响, 以及异常检测系统主动适应漂移的必要性.

2) 在系统的特征提取部分, 提出了基于混合语言模型的特征提取方法, 解决日志文本中因概念漂移导致的“词汇外”问题, 避免出现因无法识别而无法提取特征的情况.

3) 在系统的检测部分, 提出了一种基于极值理论的持续学习方法应对 Web 访问日志的概念漂移所导致的检测模型性能下降问题, 实现异常检测系统的主动适应, 以使检测性能保持平稳.

2 相关工作

2.1 网络攻击检测

相比于基于特征匹配的方法, 基于机器学习的方法对 Web 攻击检测更加有效[8]. 支持向量机 (SVM) 的变种由于其高准确性和效率, 成为 Web 攻击检测的主要方法[9,10]. 其他基于机器学习的方法, 如稀疏向量分解[11]、隐马尔可夫模型[12]和轻量梯度提升机[13]在检测方面也表现优异. 然而, 这些方法都需要选择合适的特征进行学习, 选择的特征对模型性能有显著影响.

深度学习方法通过训练多层神经网络来学习数据的有效表示, 减轻特征选择的负担. 此外, 在处理复杂攻击时, 深度学习方法在精确率和效率上已被证明优于机器学习方法[14,15]. 流行的方法包括卷积神经网络 (CNN)[5,16]、长短期记忆网络 (LSTM) 等, 为 Web 攻击检测提供了高精确率的解决方案[17,18].

无论是基于机器学习还是深度学习的方法, 在实际应用中模型都面临在时间上数据漂移的敏感问题.

具体来说, 随着攻击手段的发展, 实时数据流中可能会出现新型和未知的攻击. 之前训练良好的方法可能无法检测到这些攻击. 为了弥补这一差距, 在现有的准确检测方法的基础上, 我们专注于高效的模型适应方法, 以应对数据的漂移.

2.2 数据漂移检测与适应

在 Web 攻击检测研究中, 数据漂移可以分为两种情况: 1) 小变化: 已知类别的模式在测试集中, 显著变化; 2) 大变化: 在训练模型中出现的未知模式[4]. Wang 等人[19]使用大语言模型微调策略捕捉上下文信息, 利用小样本数据构建模型并检测数据漂移. 然而, 该模型无法检测剧烈的数据漂移. Xie 等人[20]设计了 Multi-CAD 方法, 通过 p 值作为日志数据漂移检测的指标. Yang 等人[4]构建了一个基于对比学习的自编码器, 以实时检测漂移样本, 并使用绝对中位数偏差作为距离函数解释漂移. 然而仅检测数据漂移是不够的, 最终目标是使之前的模型适应这种数据漂移. Jain 等人[21]结合滑动窗口和 K-means 重新训练模型, 从而保持其在数据流中异常检测的性能. Andresini 等人[3]提出了一种基于增量学习的持续更新的入侵检测方法, 并在网络流量数据集上测试. Kan 等人[22]解决了数据漂移适应方法 DroidEvolver[23]的一些缺点, 例如, 由于低质量伪标签导致的性能下降, 并提供了一个鲁棒性更强的变体用于恶意软件检测. 已有的一些数据漂移适应研究文献并不完全适用于 Web 攻击检测的场景. 例如, Web 攻击检测是文本数据, 不能通过统计方法完全表示. 因此, 我们设计提出了一种用于 Web 攻击的异常检测系统, 该系统能够有效地从文本数据中提取特征, 更新检测模型, 以减轻数据漂移的影响.

3 方法设计

本节将介绍所提出的基于持续学习策略的 Web 异常检测系统, 持续克服 ELSV 概念漂移的问题, 实现检测模型的增量更新, 适应日志数据中出现的新的异常类别或者新的正常类别.

3.1 系统概括

本文所提出的基于持续学习策略的 Web 异常检测系统如图 1 所示, 主要包括特征提取模块和异常检测模块. 从 Web 服务器收集到日志文本后, 特征提取模块将日志文本向量化, 随后在异常检测模块检测.

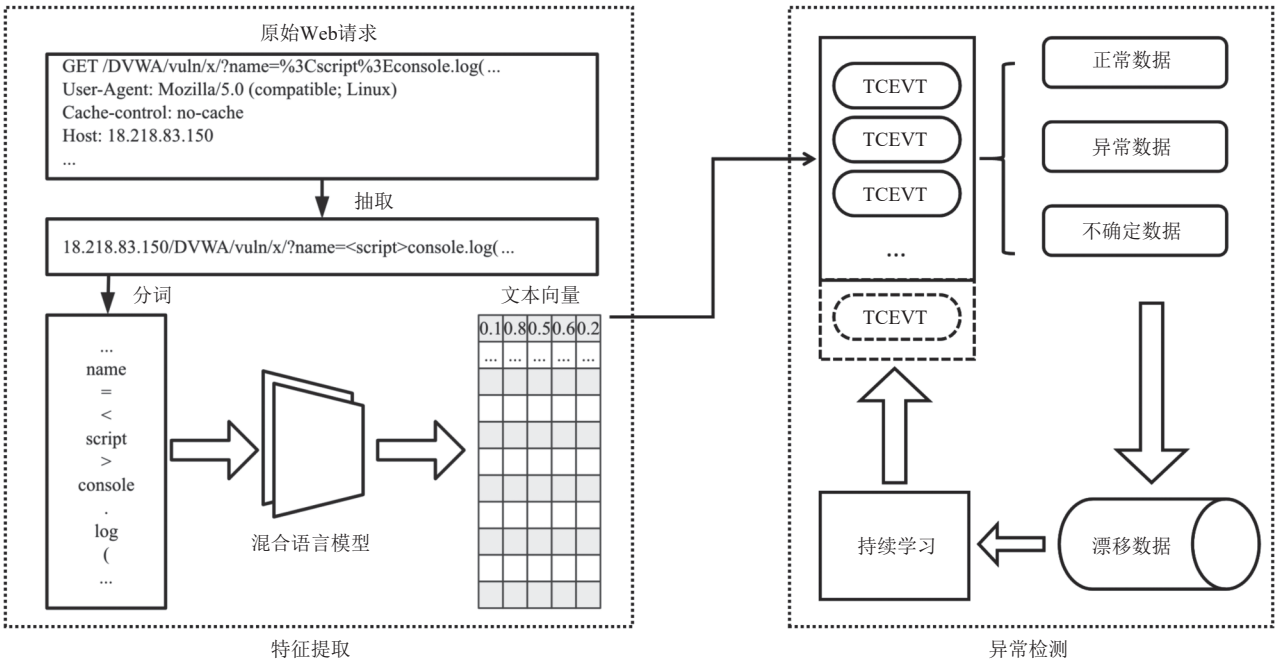


图 1 基于持续学习策略的 Web 异常检测系统

图 1 中, 在特征提取模块提出了一种混合语言模型的特征提取方法, 可以更全面的获取日志文本中的信息. 图 1 中的特征提取模块展示了从原始日志经过提取关键文本、文本分词, 最后通过语言模型向量化得到特征矩阵的过程. 同时, 在异常检测模块我们设计了一种持续学习策略, 使异常检测器能不断地学习和

更新, 以适应数据漂移. 图 1 中基于持续学习的异常检测包含将日志特征向量矩阵经过集成检测器检测、并将漂移样本送入模型进行持续学习以更新异常检测模型的迭代循环过程.

3.2 基于混合语言模型的特征提取方法

HTTP 是文本形式的传输协议, 将其转换为数值数据需要对数据结构化. 根据 HTTP 协议的结构, 从原始日志中提取 Host、URL 等信息. 具体通过文本中的特殊字符分割对文本进行分词. 每个字段由若干词组成“句子”. 再通过文本处理的方法进行向量化处理将每个“句子”转换为数值向量.

在现有的日志异常检测研究中, 向量化处理常用以单词 (word) 为基本单位的语言模型, 如 Word2Vec 和 GloVe 等, 虽然能很好地提取和表示词的语义信息, 但这两种方法都无法处理词汇外 (out-of-vocabulary) 问题. 该问题在较短的日志数据集上影响较小, 但对于时间跨度更长的现实数据来说, 随着时间的推进, 词汇外词会越来越多, 如果不解决, 会导致所有的词汇外词表示为同一向量, 影响样本的预测. 为解决该问题, 相关研究提出了以字符 (character) 或子词 (sub-word) 为单位的语言模型, 但是这两种模型都在不同程度上损失了原本词的语义信息. 因此, 本文提出了一种基于混合语言模型的日志词向量化方法, 结合使用词语言模型和子词语言模型, 使两者优势互补, 处理 Web 日志异常检测场景中的文本特征提取.

在语言模型预训练阶段, 包含两个模型: 词级模型使用 Word2Vec 中的 skip-gram 模型, 并将输入的日志切分为单词, 从而学习日志中词的语义信息; 子词级模型使用同样的 skip-gram 模型结构, 但其输入使用日志文本中单词进一步切分的 n-gram 子词, 从而在子词水平训练并提取信息, 一旦有词汇外词, 便可利用子词模型对该单词生成相应的词向量.

图 2 为日志向量化方法. 在日志向量化阶段, 对于每条经过分词后的日志中每一个词执行如下步骤.

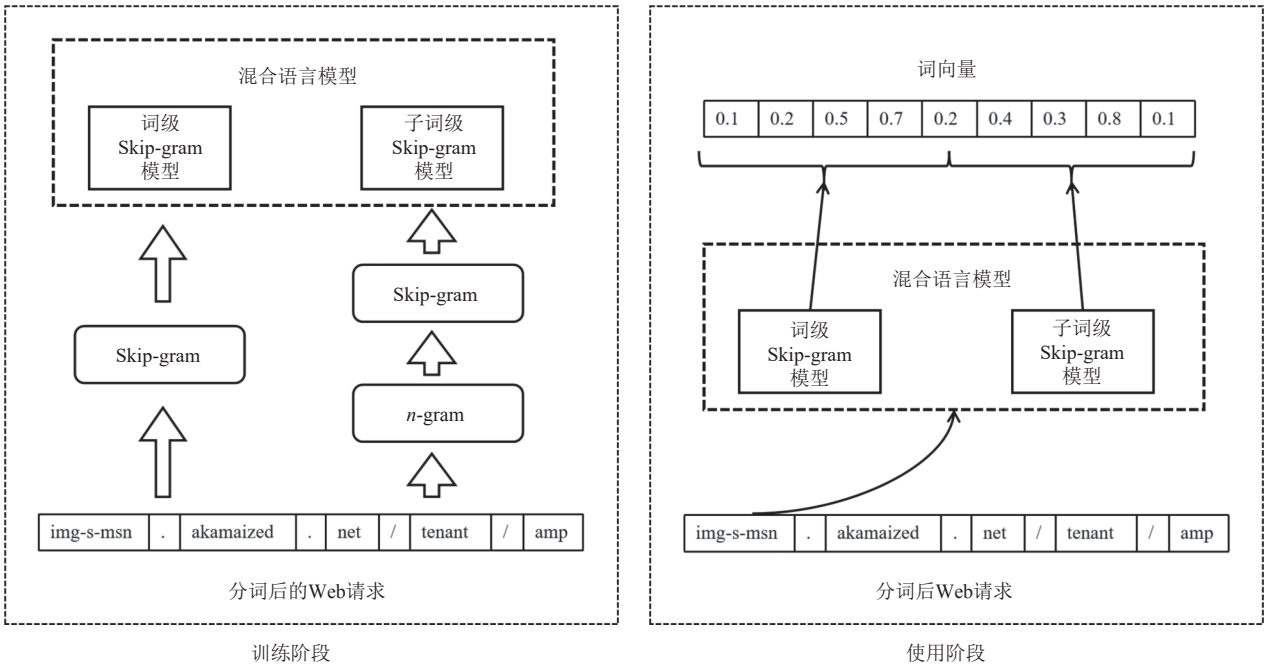


图 2 日志向量化方法

1) 定义词级模型, 使用分词语料完成训练得到词汇表$vocab_A$; 定义子词级模型, 使用 n-gram 语料完成训练得到子词表$vocab_B$;

2) 初始化W的词向量$S=[s_1,s_2,\cdots,s_n]$;

3) 查询$vocab_A$中是否存在W, 若存在, 则获得其词向量$S'=\left[s'_1,s'_2,\cdots,s'_{2/n}\right]$, 并替换$S$中前$2/n$个值, 若不存在, 则继续步骤 4);

4) 将W按照 n-gram 拆分为若干子词$[w_1,w_2,\cdots,w_m]$, 然后分别对子词w_i在$vocab_B$中获取向量表示, 最终将所有子词得到的向量求和得到$S'=\left[s''_1,s''_2,\cdots,s''_{2/n}\right]$, 并替换$S$中后$2/n$个值;

完成词W的词向量S的计算. 由此, Web 原始日志

由原来的文本变换为了数值向量形式.

3.3 基于极值理论的持续学习异常检测方法

文献[2]使用 TextCNN, 一种用于文本处理任务的卷积神经网络, 对输入文本的词嵌入矩阵应用一维卷积层来处理, 使用多个卷积核捕捉不同的局部特征, 随后通过最大池化层提取最显著的特征. 通过 TextCNN 处理 Web 日志的文本输入, 可以提取文本结构中的空间特征, 更好地捕捉局部相关性. 为了适应概念漂移的日志数据, 基于这个模型我们提出了一种基于极值理论的持续学习异常检测方法. 其中利用极值理论 (extreme value theory, EVT) 对数据中的极值进行分析, 用来处理概率分布中位数的极差. 它描述了接近极限时极端值的分布. Scheirer 等人[24]在研究中提出预测分数的分布尾部包含重要的信息.

我们利用尾部信息来判别偏离已知分布的点. 参考 Bendale 等人[25]在公开数据集使用 EVT 异常检测的思路, 我们使用 EVT 检测偏离已有模型学习到的数据分布样本, 筛选出异常检测模型不能确定的样本, 作为持续学习更新步骤中进行训练的样本, 避免使用全量样本更新模型带来的高成本. 在初始化阶段, 由于本文异常检测模型是若干子模型的集合, 所以使用标记好的日志样本训练第 1 个子模型m_0. 另外, 引入极值理论为正常/异常类别分别拟合一个分布得到wd_0来计算输入样本. 算法 1 给出了通过极值理论拟合分布的详细步骤: 首先获取训练数据在训练好的 TextCNN 模型中倒数第 2 层的输出向量$V(x)$; 然后对正常/异常类型样本对应的激活向量分别计算平均向量得到μ_{normal}和μ_{abnormal}; 其次对正常/异常样本对应的激活向量分别计算与其平均向量的距离; 最终基于这些距离分别对正常样本/异常样本使用 libMR[24]拟合 Weibull 分布.

算法 1. TextCNN 拟合极值理论 (EVT) 模型

输入: $V(x)$的输出向量、训练集X_{normal}和X_{abnormal}、libMR 的参数η.
输出: μ_{normal}、μ_{abnormal}、libMR 模型wd_{normal}和wd_{abnormal}.

1. 计算X_{normal}的$V(x)$平均值, $\mu_{\text{normal}}=\text{mean}(V(X_{\text{normal}}))$
2. 为正常样本拟合 EVT, $wd_{\text{normal}}=\text{libMR}.FitHigh(\|X_{\text{normal}}-\mu_{\text{normal}}\|, \eta)$
3. 计算X_{abnormal}的$V(x)$平均值, $\mu_{\text{abnormal}}=\text{mean}(V(X_{\text{abnormal}}))$
4. 为异常样本拟合 EVT, $wd_{\text{abnormal}}=\text{libMR}.FitHigh(\|X_{\text{abnormal}}-\mu_{\text{abnormal}}\|, \eta)$
5. 返回μ_{normal}、wd_{normal}、μ_{abnormal}和wd_{abnormal}

算法 2 介绍了子模型m_i通过拟合的分布wd_i重新计算输出的过程: 对于输入的样本, 首先获得其经过 TextCNN 模型映射后的激活向量; 然后, 将这个向量分别与μ_{normal}和μ_{abnormal}计算距离, 利用得到的距离放入到算法 1 中拟合的两个概率分布中计算该样本属于相应分布的概率; 最后利用该概率计算得到修正后的分类. 最终除了获取模型对样本类型的预测外, 还会得到一个样本不属于已知分布的概率. 重新计算后的预测类别若为 2, 说明输入的样本不属于该子模型m_i学习到的正常和异常中的任一分布. 从而能让子模型拒绝预测自己不确定的样本, 以此得到异常检测模型更新时所需要的样本. 实现过程中, 继续使用 TextCNN 模型作为基础的子模型m_i, 训练完成后, 基于极值理论拟合出该子模型对训练样本的映射分布wd_0, 由此结合得到可更新子模型 TCEVT (TextCNN with EVT), 加入异常检测模型的集合 (图 3 中的检测器), 完成异常检测模型的更新和扩充.

算法 2. 重计算分类结果

输入: μ_{normal}、μ_{abnormal}、libMR 模型wd_{normal}和wd_{abnormal}、训练集X_{test}.
输出: X_{test}的分类结果Y'_{test}和异常分数S_{test}.

1. for x_i in X_{test} do
2. 计算平均$v_i(x_i)$
3. $wscore_{\text{normal}}=wd_{\text{normal}}\cdot wscore(\|v_i-\mu_{\text{normal}}\|)$
4. $wscore_{\text{abnormal}}=wd_{\text{abnormal}}\cdot wscore(\|v_i-\mu_{\text{abnormal}}\|)$
5. $v'_i[0]=(1-wscore_{\text{normal}})\times v_i[0]$, $v'_i[1]=(1-wscore_{\text{normal}})\times v_i[1]$
6. $Y'_{\text{test}}.append(\text{argmax}([v'_i[0], v'_i[1], 1-v'_i[0]-v'_i[1]]))$
7. $S_{\text{test}}.append(\max([v'_i[0], v'_i[1], 1-v'_i[0]-v'_i[1]]))$
8. 返回Y'_{test}和S_{test}

由于本文所提出的基于持续学习策略的异常检测方法由若干子检测器构成, 即集成了若干 TCEVT 模型, 称为 E-TCEVT. 并随着异常检测模块的持续学习, TCEVT 模型数量还会不断增加. 图 3 中的持续学习阶段过程可以描述为以下步骤.

1) 定义训练的模型 E-TCEVT 作为异常检测模型, 定义X_i为第i个批次到达的待检测数据;

2) 将待检测数据X_i输入 E-TCEVT 中每个 TCEVT 中, 每个 TCEVT 根据算法 2 计算检测结果, 得到正常/异常/不确定, 及其相应的概率得分;

3) X_i中每个样本的最终检测结果为: a) 若所有 TCEVT 将其检测为不确定, 则该样本标识为不确定; b) 在所有检测结果中, 排除识别为不确定的结果后, 概率得分最高的结果作为该样本的最终检测结果.

4) 将X_i中标识为不确定的样本集称为X_{rt}, 当X_{rt}的数量超过阈值th时, 触发更新机制, 使用算法 3 更新 E-TCEVT 模型, 否则, 返回步骤 1).

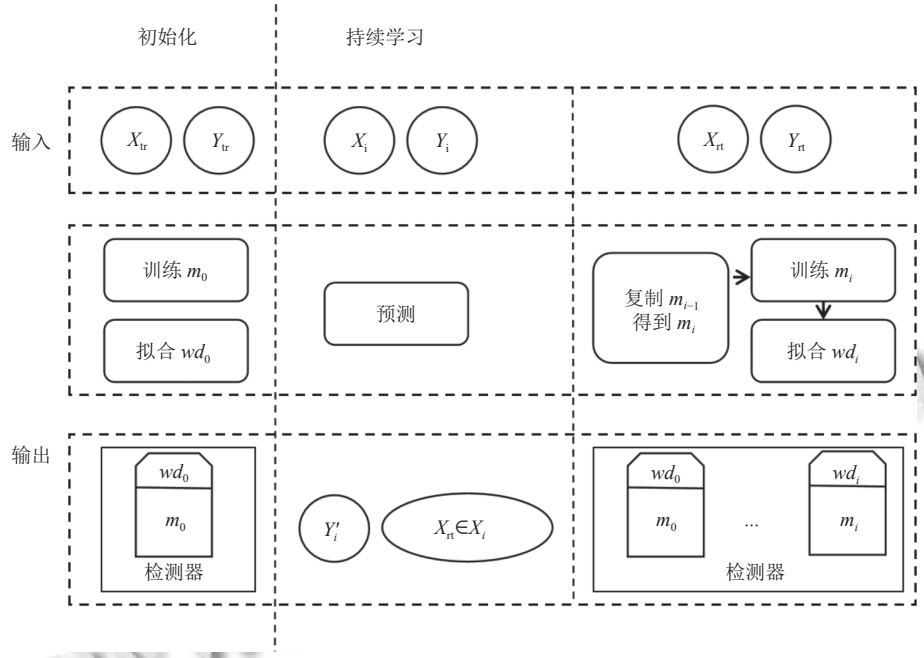


图 3 基于持续学习策略的异常检测方法框架

算法 3. 更新集成 TCEVT

输入: 标签为Y_{rt}的X_{rt}样本, 集成 TCEVT M.
输出: 更新的集成 TCEVT M.

1. 得到M最新的子模型m_n
2. 复制m_n来得到m_{n+1}
3. 用X_{rt}和Y_{rt}训练m_{n+1}
4. 用算法 1 对m_{n+1}进行调整并加入M中
5. 返回M

至此, 我们实现了通过为新的未知类型样本添加新的子检测器达到对不断漂移数据的主动适应过程. 由于保留了旧的分类器, 所以系统减少了对原本已知类型的检测性能损失; 同时也避免了通过保留历史样本的方式来平衡重新训练阶段的模型遗忘和更新.

4 实验评估

4.1 数据集

CICIDS2017 数据集[26]来自加拿大网络安全研究所 (CIC), 提供了不同设备和操作系统的完整网络测试平台的网络数据包采集. 其中 2017 年 7 月 6 日上午包含了 3 种 Web 攻击, 使用该数据进行实验.

CSE-CIC-IDS2018 数据集[27]是通信安全机构 (CSE) 与加拿大网络安全研究所 (CIC) 收集, 基于创建用户概要文件来生成用于入侵检测的多样且全面的基准数据集. 该数据集中 2018 年 2 月 22 日和 2 月 23 日包含 Web 攻击数据, 使用该数据进行实验.

实验还使用了中国气象局网站群系统的 Web 访问日志中 2024 年 7 月 23 日–7 月 29 日为期一周的真实数据, 并由专家对异常数据进行打标. 其中, 异常日志共 40434 条, 共有 8 种不同的攻击类型.

4.2 实验配置

4.2.1 实验数据集

对于开源数据集, 因为时间跨度过小, 从时间维度上切分难以体现其数据存在概念漂移, 根据对于安全领域概念漂移的定义, 存在两种情况: 一方面是已知类型中出现了样本的演化, 即已有分类的数据分布出现漂移; 另一方面是出现了新的分类.

实验中通过在测试集中添加“演化”的异常样本来模拟存在概念漂移的数据, 即将其中的某种攻击数据取出当作演化后的未知异常类型, 按照不同的比例添加到测试集中, 从而构造出漂移数据. 实验中, 基于两个数据集构造了 4 个测试集, 其中未知异常的占比分别为 0%、25%、50% 和 75%, 分别为测试集 A、B、C 和 D. 表 1 为两个数据集中各类样本的数量统计, 由于两个数据集中关于 SQL 注入的攻击样本过少, 因此这里使用 XSS 和暴力破解 Web 两种攻击作为异常样本, 并在训练集使用其中一种, 另一种则作为未知异常的样本, 测试集按照上述 4 种比例构造.

对于真实数据集, 时间跨度比较大, 所以可以从时

间维度上划分, 利用较早时间的数据训练模型, 后续时间的数据切分若干时间段分别进行测试, 观察在真实场景中模型的性能漂移情况.

表 1 数据集中各类样本的数量

数据集	正常	暴破	XSS	SQL注入
CICIDS2017	47031	7270	1824	12
CSE-CIC-IDS2018	1211493	13119	11174	51

4.2.2 对比算法

首先, 为了对比和验证非更新的模型是否都存在性能随数据集的分布差异漂移而下降的问题, 我们选用了以下几种模型作为对比算法: 多层感知机 (MLP) 作为人工神经网络 (ANN)[16]的一种形式, 以其能够学习输入数据中复杂的非线性模式的优势, 特别适用于需要捕捉大量历史数据中潜在关系的日志分析任务; 专为文本数据分析适配的一维卷积神经网络 (CNN)[4,14], 通过其卷积层有效提取局部特征, 例如在日志数据中常见的重复模式和关键字, 展现出优异的处理性能; 循环神经网络 (RNN)[17,18], 能够处理具有时间序列特性的 Web 日志数据, 通过记忆以前的信息, 更好地理解数据流中的时间动态; 语义向量化的集成学习分类 (ELSV), 通过集成多个模型来提高整体的预测准确性和稳定性, 特别适合于处理 Web 日志数据, 因为它可以从多个模型学习到的不同特征中获益, 提高对新类型攻击的检测能力. 这些模型各自展示了在处理文本数据时的不同技术和方法, 以全面评估本文方法的效果和优势. 另外, 也参考了 INSOMNIA[3], 测试了在单个 TCEVT 模型上使用微调方法时的性能表现.

其次, 为了验证本文所提出的基于混合语言模型的特征提取方法的先进性, 在实验中对比了词模型、子词模型结合本文的异常检测模型达到的性能表现.

4.2.3 评估指标

度量训练集和测试集之间的分布差异, 使用最大平均差异 (maximum mean discrepancy, MMD). 该方法用于判断两个分布 p 和 q 是否相同. 它的基本假设是: 对于所有以分布生成的样本空间为输入的函数 f, 如果两个分布生成的足够多样本在 f 上对应值的均值都相等, 那么可以认为这两个分布是同一个分布. 实验中除了传统的召回率和 $F1$ 分数外, 还选择了 AUT[27]来度量上述性能在时间尺度上的衰减, 即根据时间节点绘制性能得分曲线, 计算曲线下的面积, 用以在真实数据集上度量模型的表现.

4.3 结果分析

4.3.1 对比实验

在两个开源数据集上, 按照所述方法分别构造 4 个测试集, 它们与训练集的分布差异程度使用 MMD 衡量. 都采用词模型提取特征作为基线. 表 2 和表 3 提供了两个数据集的各测试集经过特征提取后通过 MMD 计算的分布差异, 从 MMD 值来看, CSE-CIC-IDS2018 各测试集的 MMD 值几乎是 CICIDS2017 各测试集的两倍, 这表示在 CSE-CIC-IDS2018 数据集上构造的各测试集与训练集之间的分布差异更大.

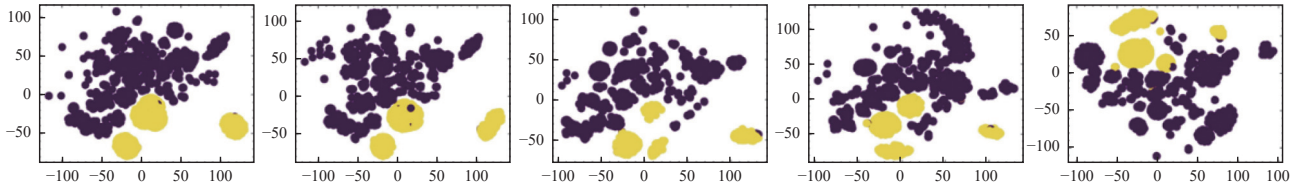
表 2 CICIDS2017 数据集

指标	测试集A1	测试集B1	测试集C1	测试集D1
MMD	0.0042	0.0071	0.0185	0.0380

表 3 CSE-CIC-IDS2018 数据集

指标	测试集A2	测试集B2	测试集C2	测试集D2
MMD	0.0177	0.0192	0.0344	0.0637

图 4 和图 5 给出了两个数据集上各数据子集的分布情况, 可以直观地看出在 CICIDS2017 数据集上构造的各测试集与训练集之间的分布差异明显小于 CSE-CIC-IDS2018 数据集上的各测试集与训练集之间的差异. 造成这一现象的根本原因在于 CICIDS2017 中两种攻击的样本都很少且网络拓扑结构较简单, 导致正常样本与异常样本间的差异明显, 时间跨度较小, 不同攻击样本之间也存在较高相似性; 而 CSE-CIC-IDS2018 所采集的网络环境更为复杂且与真实网络环境相似, 异常样本来自两天的时间跨度, 样本间的差异性更高.



(a) 训练集 (b) 测试集A1 (c) 测试集B1 (d) 测试集C1 (e) 测试集D1

图 4 CICIDS2017 训练集与测试集分布

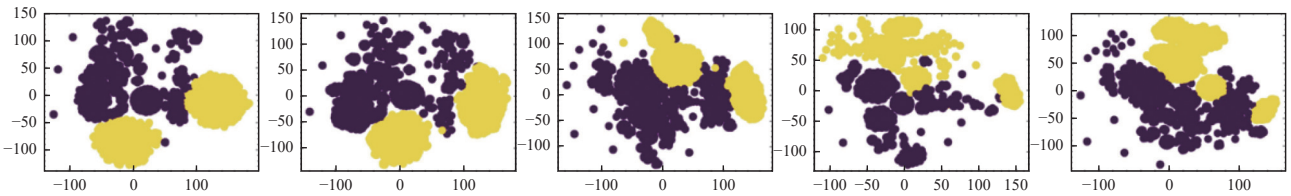


图 5 CSE-CIC-IDS2018 训练集与测试集分布

表 4 和表 5 和体现了各种模型在构造的各个测试集上的召回率和 $F1$ 分数指标, ELSV、ANN、CNN、RNN 这 4 种非更新的模型, 在面对不同程度的漂移测试集时会出现不同程度的性能下降, 并且随着与训练集分布差异的变大性能逐渐降低, 图 6 对不同模型在两个数据集上的召回率和 $F1$ 分数指标进行了可视化分析. 在 CICIDS2017 数据集上, ELSV、ANN、CNN 这 3 种模型的召回率在测试集 D1 上下降了 37%, $F1$ 分数下降了 22%, 而 RNN 分别下降了 49% 和 38%; 在 CSE-CIC-IDS2018 数据集上, 在测试集 D2 上下降的幅度更大, 模型 ANN 和 RNN 的召回率甚至下降了 74%, $F1$ 分数下降了 59%; 表现较好的 CNN 模型也分别下降了 56% 和 41%. 这几种程度的性能下降在实际使用中都是难以接受的, 因此需要模型能够对漂移的数据进行更新, 减少模型的性能下降, 以保持其在长期使用中的可用性.

表 4 CICIDS2017 上各模型性能 (%)

模型	测试集A1		测试集B1		测试集C1		测试集D1	
	召回率	$F1$分数	召回率	$F1$分数	召回率	$F1$分数	召回率	$F1$分数
ELSV	99.93	99.77	87.26	93.03	74.90	83.50	62.65	76.86
ANN	99.85	99.75	86.55	92.59	74.05	84.99	62.35	76.59
CNN	99.70	99.80	88.48	81.15	74.15	84.13	62.52	76.86
RNN	**99.96**	99.52	87.75	91.53	**75.30**	83.36	50.71	61.90
TCEVT	87.78	80.41	**90.77**	**93.68**	71.21	83.39	**63.56**	77.32

表 5 CSE-CIC-IDS2018 上各模型的性能 (%)

模型	测试集A2		测试集B2		测试集C2		测试集D2	
	召回率	$F1$分数	召回率	$F1$分数	召回率	$F1$分数	召回率	$F1$分数
ELSV	99.97	**99.98**	80.01	**88.76**	55.12	70.57	28.87	43.82
ANN	99.90	99.95	74.90	85.64	50.05	66.71	25.35	40.06
CNN	99.93	99.89	74.88	85.61	60.03	74.31	43.87	58.56
RNN	**99.98**	97.97	74.95	83.00	49.95	62.83	25.15	40.19
TCEVT	**99.98**	94.65	**93.75**	74.27	**95.11**	65.20	**85.03**	72.55

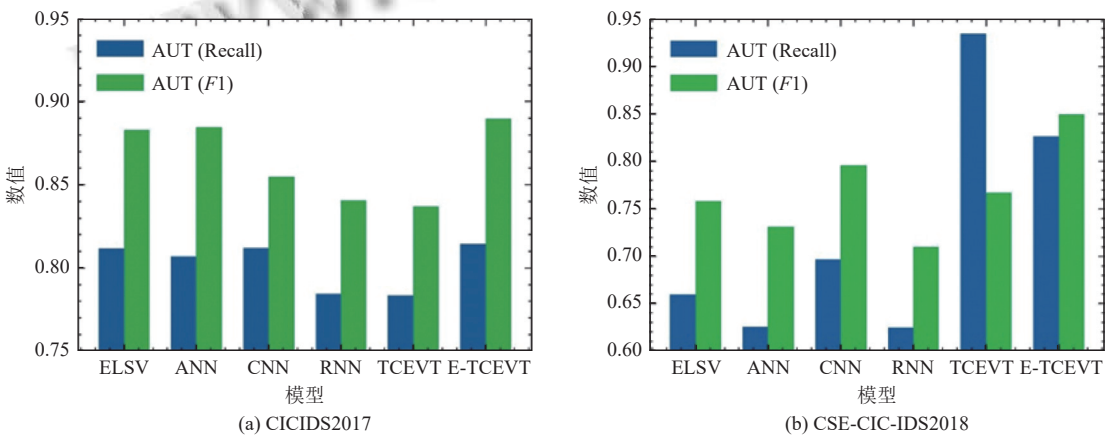


图 6 AUT 指标表现

4.3.2 E-TCEVT 实验

为了观察基于混合语言模型的特征提取方法性能表现, 我们对比了基于词模型、子词模型及混合语言模型的特征提取方法对于 E-TCEVT 持续学习模型的性能影响, 从表 6 和表 7 可以看出基于混合语言模型的特征提取方法对比单一语言模型的特征提取方法, 对后续异常检测模型的性能有积极的作用, 基于混合模型的方法普遍在 *F*1 分数上表现更好. 从图 7 中也可以直观地看出基于混合语言模型的 E-TCEVT 具有更高的 AUT 指标, 在 CICIDS2017 数据集和 CSE-CIC-IDS2018 数据集上的 AUT (召回率) 分别达到了 83.86% 和 83.37%, 而 AUT (*F*1 分数) 指标分别达到了 90.51% 和 89.49%, 这两项指标都高于其他两种基于单一语言模型的指标.

表 6 CICIDS2017 上向量化模型对比 (%)

模型	测试集A1		测试集B1		测试集C1		测试集D1	
	召回率	*F*1分数	召回率	*F*1分数	召回率	*F*1分数	召回率	*F*1分数
Word-E-TCEVT	99.92	99.87	87.39	93.13	75.28	85.37	63.23	77.59
Subword-E-TCEVT	84.84	87.69	80.68	87.03	68.75	78.33	62.86	75.08
Hybrid-E-TCEVT	**99.95**	**99.97**	**88.50**	**93.87**	**76.32**	**86.43**	**70.65**	**81.76**

表 7 CSE-CIC-IDS2018 上向量化模型对比 (%)

模型	测试集A1		测试集B1		测试集C1		测试集D1	
	召回率	*F*1分数	召回率	*F*1分数	召回率	*F*1分数	召回率	*F*1分数
Word-E-TCEVT	99.94	**99.97**	81.18	83.07	**82.80**	83.35	66.61	73.30
Subword-E-TCEVT	99.98	99.89	77.17	85.21	75.43	80.40	56.18	66.56
Hybrid-E-TCEVT	**99.99**	**99.97**	**88.95**	**93.30**	75.55	**86.07**	**71.00**	**78.63**

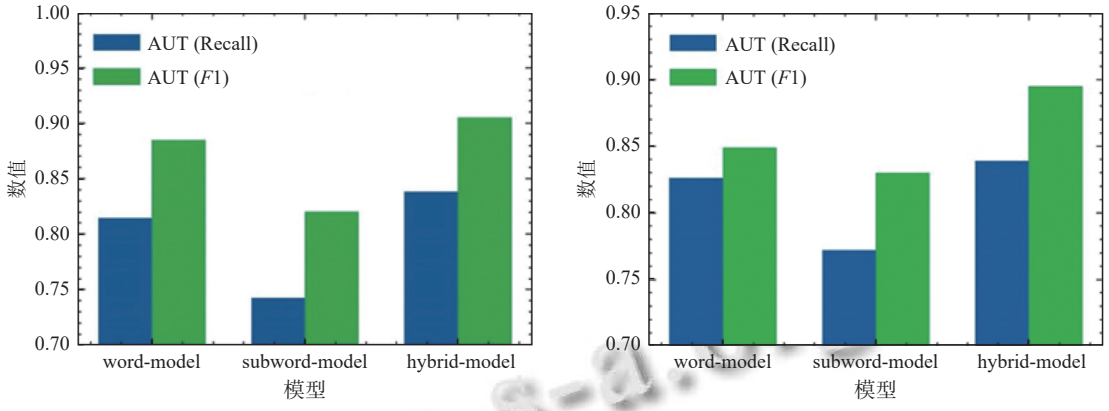


图 7 语言模型对 E-TCEVT 的性能影响

另外, 从词汇外词 (OOV) 的角度分析, 如图 8 所示, 本文所用的混合模型与词模型相比, 在提取特征后计算 MMD 值时几乎表现一致, 在含有词汇外词的样本较少时混合模型由于生成的特征维数更高, 会在计算 MMD 值时较词模型更高; 而在含有词汇外词的样本较多时, 由于混合模型会使用子词模型中的子词信息, 混合模型生成的特征在计算 MMD 值时较词模型更低. 可以看出词汇外词也在一定程度上影响了测试集与训练集的分布差异.

在表 4 和表 5 中, E-TCEVT 及其子模型 TCEVT 相较于其他非更新模型在召回率和 *F*1 分数上都有不同程度的提升. 实验中, E-TCEVT 基于多模型集成方法, 在每次更新中添加新的子模型; TCEVT 方法则是使用微调的方式在模型自身上更新. 从表 4 和表 5 的实验数据可以看出, TCEVT 方法在两个数据集上都表现出了较高的召回率, 在 CICIDS2017 数据集上相较于之前的 ELSV 高出了 1%–3%; 而在 CSE-CIC-IDS2018 数据集上该指标相较于 ELSV 提升了 13%–56%. E-TCEVT 方法则在 *F*1 分数上表现更好, 在 CICIDS2017 数据集上相较于之前的 ELSV 高出了 1%–2%; 而在 CSE-CIC-IDS2018 数据集上该指标相较于 ELSV 提升了 13%–30%. 这种结果与前文所分析的两个数据集

上构造的测试集与训练集的差异一致, 在 CSE-CIC-IDS2018 数据集上的各个测试集与训练集差异更大, 对于非更新模型的挑战也更大, 因此更新的模型提升也会更大.

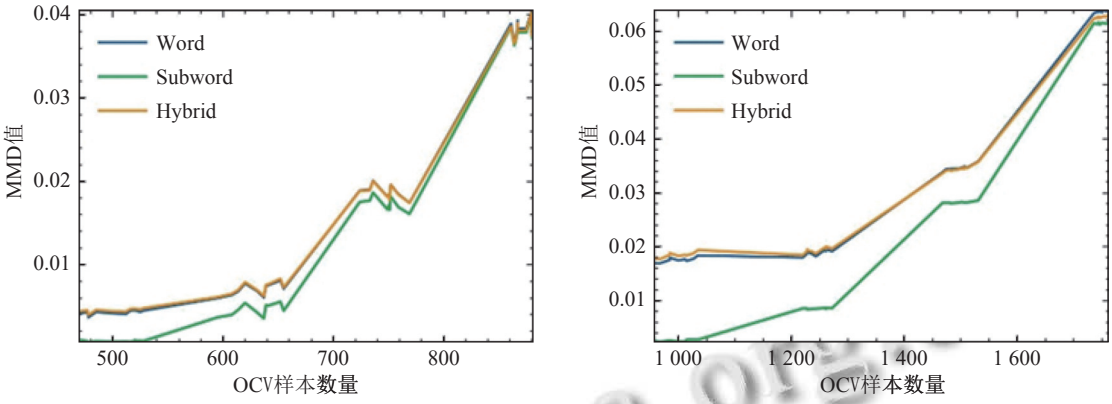


图 8 各语言模型上 MMD 随 OOV 数量的变化

另外, E-TCEVT 相较于 TCEVT 在 $F1$ 分数上的优势也体现了单一模型进行微调实现持续学习的弊端. 如果使用差异较大的新样本对模型进行微调时, 会导致模型遗忘学习到的旧样本, 对旧样本的表现下降. 由于添加到测试集中的未知样本都是恶意样本, 因此更新时更多地学习的是恶意样本, 也就导致了召回率提高而 $F1$ 分数反而有下降. 但 E-TCEVT 综合了所有子模型的输出来作为最终预测, 从而保留了对旧样本类型的检测性能, 以达到更均衡的性能表现.

为了在真实数据上评估我们的方法, 我们收集了 2024 年 7 月 23 日–7 月 29 日为期一周的网络流量日志数据, 由专家标记出异常样本. 最终共采集 40 434 条样本. 并对其中的 IP 地址进行数据增强, 调用第三方应用扩展出国家、城市、运营商、经度和纬度特征. 测试时按照未知异常占比分别为 0%、25%、50% 和 75%, 构建测试集 A、B、C 和 D. 表 8 显示了 E-TCEVT 在真实数据集上的表现, 在测试集 A 上精确率上达到 98.55%, 召回率达到 99.27%, $F1$ 分数达到 98.91%.

表 8 真实流量数据集检测结果 (%)

数据集	准确率	精确率	召回率	$F1$分数
测试集A	99.27	98.55	99.27	98.91
测试集B	86.45	79.48	98.28	87.88
测试集C	82.30	75.40	90.20	82.00
测试集D	70.55	68.40	66.80	73.10

5 结论与展望

我们基于 ELSV 在真实网络环境中遇到的测试集与训练集分布情况不一致导致检测器性能下降的问题, 设计了一种基于持续学习策略的异常检测系统 TCEVT, 针对 Web 日志数据的文本向量化方法在面对显示数据时常出现词汇外词的问题, 提出了一种基于混合语言模型的 Web 日志向量化方法; 同时基于极值理论使模型放弃未知概率较高的样本, 减少对标签的依赖; 另外, 为了解决单一模型在持续学习中对旧样本的灾难性遗忘, 我们采用了集成的思想, 保留了旧的模型, 在决策时综合所有子模型进行预测, 并且在开源数据集和真实数据上的实验都证明了所提出方法的有效性, 在 CSE-CIC-IDS2018 数据集上相较于先前的 ELSV 方法, $F1$ 分数提升了 13%–30%, 并且 AUT ($F1$ 分数) 指标达到了 89.49%, 显著高于传统非更新方法和单模型微调的方法. 在未来的工作中, 我们考虑探索其他减少人工打标签的成本且不会带来较大性能损失的方法, 从而减少系统在部署后的人力投入.

参考文献

1 Wan W, Shi X, Wei JX, *et al*. ELSV: An effective anomaly detection system from Web access logs. Proceedings of the 2021 IEEE International Performance, Computing, and Communications Conference. Austin: IEEE, 2021. 1–6. [doi: 10.1109/IPCCC51483.2021.9679413]

2 HTTP dataset CSIC 2010. https://impactcybertrust.org/dataset_view?idDataset=940. (2007-11-05) [2024-10-20].

3 Andresini G, Pendlebury F, Pierazzi F, *et al*. INSOMNIA: Towards concept-drift robustness in network intrusion detection. Proceedings of the 14th ACM Workshop on Artificial Intelligence and Security. ACM, 2021. 111–122. [doi: 10.1145/3474369.3486864]

4 Yang LM, Guo WB, Hao QY, *et al*. CADE: Detecting and explaining concept drift samples for security applications.

Proceedings of the 30th USENIX Security Symposium. USENIX Security, 2021. 2327–2344.
5 Tekerek A. A novel architecture for Web-based attack detection using convolutional neural network. Computers & Security, 2021, 100: 102096. [doi: 10.1016/j.cose.2020.102096]
6 Tan Y, Rui M, Heng W, *et al*. KnowleNet: Knowledge fusion network for multimodal sarcasm detection. Information Fusion, 2023, 100: 101921. [doi: 10.1016/j.inffus.2023.101921]
7 Ren X, Hu YP, Kuang WX, *et al*. A Web attack detection technology based on bag of words and hidden Markov model. Proceedings of the 15th International Conference on Mobile Ad Hoc and Sensor Systems. Chengdu: IEEE, 2018. 526–531. [doi: 10.1109/MASS.2018.00081]
8 Wang W, Zhang XL. High-speed Web attack detection through extracting exemplars from HTTP traffic. Proceedings of the 2011 ACM Symposium on Applied Computing. Taichung: ACM, 2011. 1538–1543. [doi: 10.1145/1982185.1982512]
9 Uwagbole SO, Buchanan WJ, Fan L. Applied machine learning predictive analytics to SQL injection attack detection and prevention. Proceedings of the 2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM). Lisbon: IEEE, 2017. 1087–1090. [doi: 10.23919/INM.2017.7987433]
10 Gogoi B, Ahmed T, Saikia HK. Detection of XSS attacks in Web applications: A machine learning approach. International Journal of Innovative Research in Computer Science & Technology, 2021, 9(1): D10962. [doi: 10.21276/ijircst.2021.9.1.1]
11 Liao Q, Li H, Kang SL, *et al*. Application layer DDoS attack detection using cluster with label based on sparse vector decomposition and rhythm matching. Security and Communication Networks, 2015, 8(17): 3111–3120. [doi: 10.1002/sec.1236]
12 Liu TL, Qi Y, Shi L, *et al*. Locate-then-detect: Real-time Web attack detection via attention-based deep neural networks. Proceedings of the 28th International Joint Conference on Artificial Intelligence. Macao: AAAI Press, 2019. 4725–4731. [doi: 10.5555/3367471.3367700]
13 Riera TS, Higuera JRB, Higuera JB, *et al*. A new multi-label dataset for Web attacks CAPEC classification using machine learning techniques. Computers & Security, 2022, 120: 102788. [doi: 10.1016/j.cose.2022.102788]
14 Rong W, Zhang BW, Lv XX. Malicious Web request detection using character-level CNN. arXiv:1811.08641v1, 2018.
15 Stevanović N, Todorović B, Todorović V. Web attack detection based on traps. Applied Intelligence, 2022, 52(11): 12397–12421. [doi: 10.1007/s10489-021-03077-9]
16 Yue T, Li Y, Hu ZH. DWSA: An intelligent document structural analysis model for information extraction and data mining. Electronics, 2021, 10(19): 2443. [doi: 10.3390/electronics10192443]
17 Tan Y, Shi XZ, Mao R, *et al*. SarcNet: A multilingual multimodal sarcasm detection dataset. Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024). Torino: ELRA and ICCL, 2024. 14325–14335.
18 Ramos Júnior LS, Macêdo D, Oliveira ALI, *et al*. LogBERT-BiLSTM: Detecting malicious Web requests. Proceedings of the 31st International Conference on Artificial Neural Networks. Bristol: Springer, 2022. 704–715. [doi: 10.1007/978-3-031-15934-3_58]
19 Wang H, Yue T, Ye X, *et al*. Revisit finetuning strategy for few-shot learning to transfer the emdeddings. Proceedings of the 11th International Conference on Learning Representations. OpenReview.net, 2023. 1–11.
20 Xie XS, Jin ZM, Wang JM, *et al*. Confidence guided anomaly detection model for anti-concept drift in dynamic logs. Journal of Network and Computer Applications, 2020, 162: 102659. [doi: 10.1016/j.jnca.2020.102659]
21 Jain M, Kaur G. Distributed anomaly detection using concept drift detection based hybrid ensemble techniques in streamed network data. Cluster Computing, 2021, 24(3): 2099–2114. [doi: 10.1007/s10586-021-03249-9]
22 Kan ZL, Pendlebury F, Pierazzi F, *et al*. Investigating labelless drift adaptation for malware detection. Proceedings of the 14th ACM Workshop on Artificial Intelligence and Security. ACM, 2021. 123–134. [doi: 10.1145/3474369.3486873]
23 Xu K, Li YJ, Deng R, *et al*. DroidEvolver: Self-evolving android malware detection system. Proceedings of the 2019 IEEE European Symposium on Security and Privacy. Stockholm: IEEE, 2019. 47–62. [doi: 10.1109/EuroSP.2019.00014]
24 Scheirer WJ, Rocha A, Micheals RJ, *et al*. Meta-recognition: The theory and practice of recognition score analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011, 33(8): 1689–1695. [doi: 10.1109/TPAMI.2011.54]
25 Bendale A, Boult TE. Towards open set deep networks. Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 1563–1572. [doi: 10.1109/CVPR.2016.173]
26 Sharafaldin I, Lashkari AH, Ghorbani AA. Toward generating a new intrusion detection dataset and intrusion traffic characterization. Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP). Funchal: SciTePress, 2018. 108–116. [doi: 10.5220/0006639801080116]
27 Pendlebury F, Pierazzi F, Jordaney R, *et al*. TESSERACT: Eliminating experimental bias in malware classification across space and time. Proceedings of the 28th USENIX Security Symposium. Santa Clara: USENIX Security, 2019. 729–746.

(校对责编: 王欣欣)