第 3 卷◆第 3 期◆版本 1.0◆2025 年
文章类型：论文|刊号（ISSN）：2972-4236(P) / 2972-4244(O)

基于 HMM-LSTM 的恶意软件行为检测方法

侯梦迪[1] 黄家广[2,*] 黄剑波[3]
1 广西机器视觉与智能控制重点实验室
2 广西高校智能软件重点实验室
3 桂林理工大学南宁分校
DOI:10.12238/acair.v3i3.15566

[摘 要] 本文提出一种基于系统调用信息的恶意软件行为检测方法,融合HMM与LSTM算法优势:利用HMM挖掘软件行为上下文依赖关系,拆分系统调用序列以解决定长截取导致的上下文信息缺失问题；借鉴LSTM情感分析思路,在LSTM中引入注意力机制强化局部特征提取,提升恶意行为预测准确率。此外,通过多批次训练动态构建阈值,基于不同训练集ROC曲线确定批次最优阈值,结合方差分析优化全局阈值,有效拓展模型适用范围并提高检测精度。
[关键词] 软件行为；隐马尔可夫模型；长短期记忆网络；网络安全
中图分类号：TN915.08 文献标识码：A

HMM-LSTM-Based Malware Behavior Detection Method

Mengdi Hou[1] Jiaguang Huang[2,*] Jianbo Huang[3]
1 Guangxi Key Laboratory of Machine Vision and Intelligent Control
2 Guangxi Colleges and Universities Key Laboratory of Intelligent Software
3 School of Computer Application, Guilin University of Technology

[Abstract] This paper proposes a malware behavior detection method based on system call information, integrating the advantages of HMM and LSTM algorithms. It employs HMM to mine the contextual dependencies of software behaviors and segment system call sequences, addressing the contextual information loss issue caused by fixed-length segmentation in traditional detection methods. Drawing on the idea of LSTM-based sentiment analysis, an attention mechanism is introduced into LSTM to enhance local feature extraction, improving the accuracy of malicious behavior prediction. Furthermore, dynamic threshold construction is achieved through multi-batch training: the optimal threshold for each batch is determined via ROC curves of different training sets, and the global threshold is optimized by variance analysis of related parameters, effectively expanding the model's applicability scope and improving detection accuracy.
[Key words] Software Behavior; HMM; LSTM; Cybersecurity

引言

隐马尔可夫模型(HMM)擅长捕捉序列间的上下文关联特性,长短期记忆网络(LSTM)在时序数据处理方面有着明显的优势,把二者结合起来能够极大地改善检测性能,这个研究中的HMM-LSTM框架利用非固定长度分段技术来取代传统的定长截取方法,从而有效地解决由于序列长度限制造成的缺失信息的问题,而且在LSTM模块当中加入了注意力机制,这样可以更好地提取局部特征,而且经过多次迭代训练之后,会动态地调整阈值参数,从实验结果来看,这种方法在行为覆盖范围以及检测准确度这两方面都优于现有的经典模型,给恶意软件识别给予了新的解决途径。

1 HMM-LSTM分析方法简介

通过HMM模型挖掘系统调用序列中系统调用的潜在关系,使用不定长的方式截取系统调用序列可以有效提高处理后系统调用序列的信息量。

模型分为模型训练和恶意行为检测两个部分,核心是恶意行为检测部分。检测部分采用LSTM对软件行为的系统调用序列进行处理,标记存在恶意行为的系统调用短序列。训练阶段,先利用训练集中的系统调用序列训练HMM模型以获取参数,使模型能依据系统调用间的隐藏信息截取不定长有意义的短序列；再

文章类型：论文|刊号（ISSN）：2972-4236(P) / 2972-4244(O)

将HMM输出作为样本训练LSTM,生成训练数据生成函数的阈值和注意力机制参数[1]。检测时,预测函数根据阈值判断系统调用序列是否存在恶意行为。

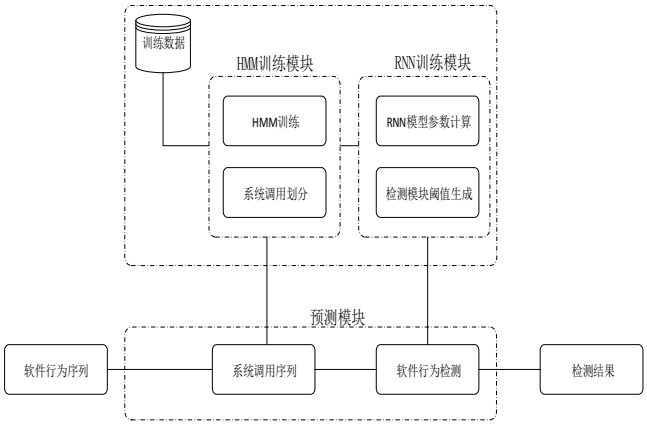

图 1

2 模型训练

2.1 HMM模型训练

HMM模型的训练的目的是根据通过已知的观测序列估计模型 λ=(A,B, π)中的参数。为了实现HMM模型对系统调用序列拆分,将预处理的一部分系统调用数据作为训练数据,通过训练估计模型参数 λ={A,B, π}。使用Baum-Welch算法作为HMM的训练算法对HMM进行训练。训练好以后使用训练好的HMM模型对系统调用序列进行拆分,使用维特比(Viterbi)算法对系统调用序列进行处理,截取符合软件行为的最佳系统调用短序列[2]。

2.2 LSTM模型训练与预测

在情绪分析模型中,模型会在接收到文本最后一个输入以后执行反向传播,所以目标函数可以用公式1表示：

$$L\left(y,y^{'}\right)=\frac{1}{2}\left\|y^{'}\left(e_{t}\right)-y\right\|_{L2} \tag{1}$$

其中 $y^{'}$ 表示预测,y表示目标。

以上算法虽然可以直接用于恶意行为检测,但是因为函数会在接收完最后一个输入以后再执行推理,这也就导致函数不能在预测结果超过阈值的情况下立即进行标记。所以需要模型在预测阶段设定阈值,只要预测结果超过阈值整个序列就被立即标记为恶意行为。根据以上可以将目标函数改为公式2的形式

$$L\left(y,y^{'}\right)=\frac{1}{2}\left\|\max_{t}y\left(e_{t}\right)-y\right\|_{L2} \tag{2}$$

其中 $\max_{t}y^{'}\left(e_{t}\right)$ 表示整个序列中的预测最大得值。一旦当前预测结果超过阈值模型就会立即对恶意行为标记。

2.3阈值计算

测试数据的不同ROC在不同的检测单元的结果也不一样,为了使阈值在检测未标记数据时达到预期的检测效果,本文使用动态方式训练模型确定阈值。该过程通过以下算法实现。

算法如下：

输入：训练模型和模型的批次

输出：给定模型的最佳阈值

Step 1. 程序建立阈值

Step 2. 设置目标FPR

Step 3. TPR设置为空

Step 4. θs设置为空

Step 5. For 每一个批处理 do

Step 6. 构建ROC曲线

Step 7. 获取对应的目标FPR'值θ'

Step 8. 根据θ'的性能给定TPR'

Step 9. θs.add(θ')//

Step 10. TPRs.add(TPR')

Step 11. Θavg=avg(θs) //求平均值

Step 12. Θstd=std(θs)//求标准偏差

Step 13. TPRavg=avg(TPRs)//求平均值

Step 14. TPRstd=std(TPRs)//求标准偏差

Step 15. θ^{*}=ΘavgKΘstd

对于每批测试数据构建ROC并计算阈值 θ ,使得使用批次下阈值的FPR和目标FPR最接近。记录阵列 θs 中保存每批次的阈值 θ ,在评估所有测试数据批次之后,我们计算 θs 阵列中记录值的平均值和标准偏差,并且计算当前时期的最佳阈值,阈值最后的计算公式如公式3所示：

$$\theta^{*}=\delta\theta+K*\sqrt{\sigma\theta} \tag{3}$$

其中 $\delta\theta$ 和 $\sqrt{\sigma\theta}$ 表示 θs 阵列的平均值和标准偏差,K通常设置为1到2之间的超参数。在最后的软件行为检测阶段,模型接收未知样本并输出行为的恶意倾向预测,如果输出的预测大于阈值 θ 则标记为恶意行为[3]。

2.4模型的整体结构

本文中的模型整体结构如图2所示,第一层为系统调用序列分割层,每个输入E会被HMM拆分成多个系统调用子序列et。第二层是词向量嵌入层,将子序列转换为one-hot编码并传输到嵌入层,Linux系统中系统调用的数量在300个左右,在2.4.4内核版本中常用系统调用共有221个,所以在词嵌入层中词嵌入矩阵设置为300×16。第三层为注意力机制层,提取系统调用序列中的重点系统调用信息。然后将信息输入LSTM中,LSTM模型会根据提取系统调用中的上下文信息,并输出系统调用序列的全局特征。最后将输出特征传递给FC层,FC层大小设置为128,FC层会使用激活函数对系统调用的倾向性给出最终预测y'。

3 实验

3.1 HMM对软件行为构建能力实验

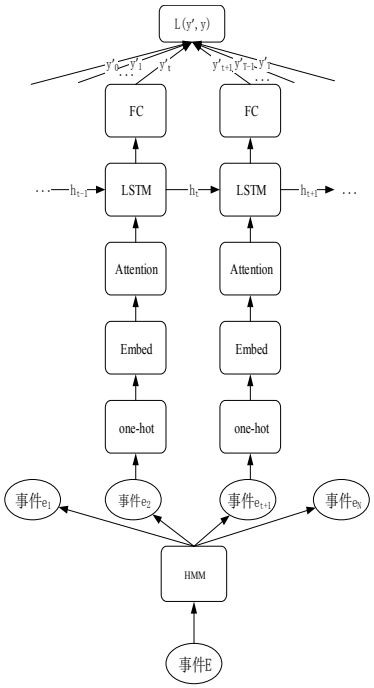

图2 模型整体结构

实验使用Alina、Mirai、Denroid和Carberp和FTP进行统计，统计生成系统短序列的大小和数量，将其和N-gram、V-gram模型相互比较。在N-gram模型中，当N的取值为6的时候检测效果较好，所以取N=6的N-gram模型进行比较，模型的比较中值比较对高频行为数量的获取情况。三个模型的比较结果如表1所示。

表1 软件行为挖掘数量对比

模型	Alina	Mirai	Denroid	Carberp	FTP
N-gram,N=6	154	184	246	346	140
V-gram	79	86	94	93	27
HMM	104	123	147	185	61

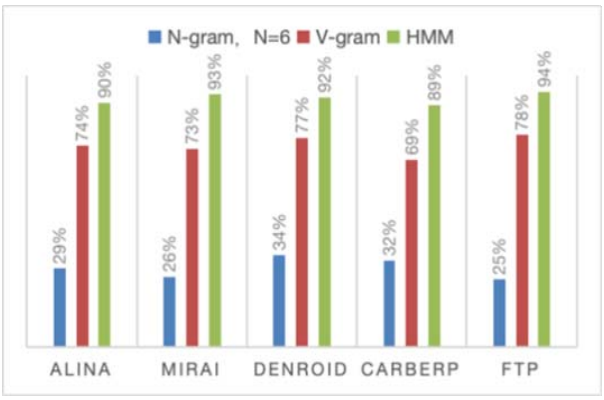

图3 软件行为覆盖率对比

表1表示三个模型对五个软件的系统调用序列的软件行为挖掘数量对比，图3表示三个模型挖掘出的软件行为在总系统调用序列中的覆盖率，通过计算挖掘出的软件行为系统调用序列在总系统调用序列中的占比得出的覆盖率，高覆盖率表示系统调用短序列更具有意义[4]。在软件行为数量方面HMM模型在挖掘出的软件行为小于N-gram模型挖掘出的数量，但是在软件行为覆盖率上HMM模型的覆盖率最高。

3.2 HMM-LSTM检测能力实验

表2 HMM-LSTM与其他检测方法对比

名称	TPR	FPR	F1值	数据集	类型
HMM-LSTM	94.4%	6.0%	0.959	16000	动态
maxNet	92.6%	4%	0.955	20000	动态
MaMaDroid	92.3%	8.6%	0.946	5500	动态
Ahmadi	86%	7.8%	0.91	1730	静态
Morales-Ortega	88.3%	10%	0.922	2700	静态
DeepClassify	92.5%	13.4%	0.939	10770	静态

表2提供了一些其他检测方法和本文提出方法的比较，其中有相关的动态、静态和基于机器学习的检测方法。和其他检测方法相比，HMM-LSTM方法在TPR方面表现最好，maxNet在FPR方面表现最好，为94.4%，F1值表示精确率和召回率的调和平均数，是对分类问题中常用的评价指标，本文HMM-LSTM模型F1值最高[5]。综合来看本文的HMM-LSTM方法检测效果最好。因为HMM-LSTM方法因为HMM处理需要花费一定的时间，在时间效率方面和其他算法比相对较弱。

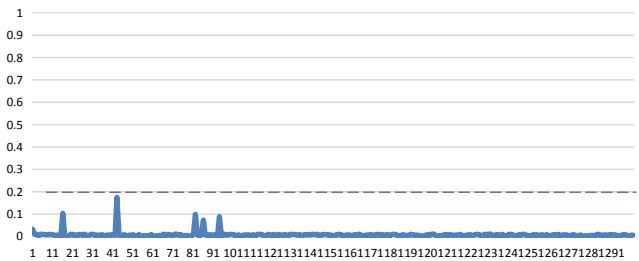

图4 良性行为测试集输出结果

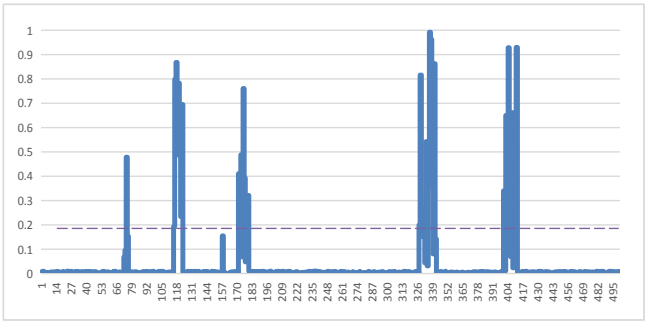

图5 存在恶意行为的测试集输出结果

最后，图4和5展示了模型对良性软件行为测试集和具有100个恶意软件行为测试集的预测效果，其中良性行为测试集长度为300个事件，恶意软件行为测试集有500个事件，两个图中阈值的位置如黄线处所标识[6-9]。如图4所示，良性软件行为的预测结

果不会超过阈值,本文的模型不会标记良性软件行为。在图5中,输入的系统调用序列55个恶意软件行为都被检测出来,这表示本文的方法可以有效检测恶意软件行为。

4 结束语

基于系统调用的恶意软件检测方法是网络空间安全领域一项重要的技术,本文提出HMM-LSTM软件行为检测方法,利用HMM基于上下文信息拆分系统调用序列,覆盖高频行为更佳;引入注意力机制的LSTMR预测恶意行为倾向,通过多批次训练动态建立阈值。实验表明,该方法较传统检测方法能更好识别恶意行为短序列。

[资助基金]

梧州学院校级青年项目(2024QN004),梧州市科技计划项目(202402021)。

[参考文献]

[1]Idika N,Mathur A P.A survey of malware detection techniques[J].Purdue University,2007,48:2007-2.

[2]陶芬,尹芷仪,傅建明.基于系统调用的软件行为模型[J].计算机科学,2010,37(4):151-157.

[3]杨宇,张健.程序静态分析技术与工具[J].计算机科学,2004,(02):171-174.

[4]蒋炎岩,许畅,马晓星,等.获取访存依赖:并发程序动态分析基础技术综述[J].软件学报,2017,28(4):745-763.

[5]Canfora G,Medvet E,Mercaldo F,et al.Detecting android malware using sequences of system calls[C]//Proceedings of the 3rd International Workshop on Software Development Lifecycle for Mobile.2015:13-20.

[6]Ferrante A, Medvet E,Mercaldo F,et al.Spotting the malicious moment:Characterizing malware behavior using dynamic features[C]//2016 11th International Conference on Availability,Reliability and Security (ARES).IEEE,2016:372-381.

[7]刘宇.基于深度学习的大词汇量连续语音识别的研究[D].重庆邮电大学,2018.

[8]Sherstinsky A.Fundamentals of recurrent neural network(RNN) and long short-term memory (LSTM) network[J].Physica D:Nonlinear Phenomena,2020,404:132306.

[9]Gronát P,Aldana-Iuit J A,Bálek M.MaxNet:Neural network architecture for continuous detection of malicious activity[C]//2019 IEEE Security and Privacy Workshops (SPW).IEEE,2019:28-35.

作者简介：

侯梦迪(1994--),男,汉族,河南平顶山人,硕士,主要研究方向为网络信息安全。

黄家广(1995--),男,汉族,广西南宁人,硕士,主要研究方向为图像处理。

黄剑波(1995--),男,壮族,广西扶绥人,讲师,硕士,目前研究方向为人工智能安全和大数据分析。