基于 BiTCN-SA 的恶意代码分类方法

黄　玮，王　坚*，吴　暄，李思聪
（空军工程大学防空反导学院，西安，710051）

摘要　当前恶意代码的对抗技术不断变化，恶意代码变种层出不穷，使恶意代码分类问题面临严峻挑战。针对目前基于深度学习的恶意代码分类方法提取特征不足和准确率低的问题，提出了基于双向时域卷积网络(BiTCN)和自注意力机制(Self-Attention)的恶意代码分类方法(BiTCN-SA)。该方法融合恶意代码操作码特征和图像特征以展现不同的特征细节，增加特征多样性。构建 BiTCN 对融合特征进行处理，充分利用特征的前后依赖关系。引入自注意力机制对数据权值进行动态调整，进一步挖掘恶意代码内部数据间的关联性。在 Kaggle 数据集上对模型进行验证，实验结果表明：该方法准确率可达 99.75%，具有较快的收敛速度和较低的误差。

关键词　恶意代码分类；特征融合；双向时域卷积网络；自注意力机制
DOI　10.3969/j.issn.2097-1915.2023.04.012
中图分类号　TP393.08　　**文献标志码**　A　　**文章编号**　2097-1915(2023)04-0077-08

A Malicious Code Classification Method Based on BiTCN-SA

HUANG Wei，WANG Jian*，WU Xuan，LI Sicong
（Air and Missile Defense School，Air Force Engineering University，Xi'an 710051，China）

Abstract　At present，the countermeasure technology of malicious code is constantly changing，and new varieties of malicious code are emerging in endless streamto make the classification of malicious code face severe challenges. Aimed at the problemsthat features extracted are insufficient and low in accuracy by using current malicious code classification methods based on deep learning，a malicious code classification method (BiTCN-SA) based on bi-directional temporal convolution network (BiTCN) and self attention mechanism is proposed. This method is combination of opcode features with image features to show different feature details，increasing feature diversity. The BiTCN is constructed to process the fused features，making full use of the pre and post dependencies of the features. The self attention mechanism is introduced todynamically adjust the data weight，further mining the correlation between the internal data of malicious code. The model is verified by using the Kaggle data set. The results show that the accuracy of this method can reach 99.75%，and the method is fast at convergence speed，lowin error，and better than the other models.

Key words　malicious code classification；feature fusion；bi-directional temporal convolution network；self attention mechanism

收稿日期：2023-02-27
基金项目：国家自然科学基金(61806219，61703426，61876189)；陕西省自然科学基金(2021JM-226)；陕西省高校科协青年人才托举计划(20190108，20220106)；陕西省创新能力支撑计划(2020KJXX-065)
作者简介：黄　玮(1999－)，男，江西赣州人，硕士生，研究方向为网络空间安全、恶意代码检测。E-mail：hw_afeu@163.com
通信作者：王　坚(1982－)，男，陕西渭南人，副教授，研究方向为智能信息处理和恶意软件检测。E-mail：26471375@qq.com

引用格式：*黄玮，王坚，吴暄，等. 基于 BiTCN-SA 的恶意代码分类方法[J]. 空军工程大学学报，2023，24(4)：77-84. HUANG Wei，WANG Jian，WU Xuan，et al. A Malicious Code Classification Method Based on BiTCNSA[J]. Journal of Air Force Engineering University，2023，24(4)：77-84.*

随着网络技术的不断发展，网络环境不断变化。国家互联网应急中心 2022 年第 24 期网络安全信息与动态周报显示[1]，2021 上半年，捕获恶意程序样本数量约 2 307 万个，均传播次数达 582 万余次，涉及恶意程序家族约 20.8 万个。随着恶意代码的肆虐，如何准确地对恶意代码进行分类已成为领域的研究热点。

传统的静态分析方法基于标签和特征库[2]，在分析已知的恶意代码方面表现良好，然而对于未知恶意代码，并不能进行很好地分析，存在一定的局限性。

由于恶意代码的种类和数量都在不断增加，恶意代码采用各种加壳、混淆等对抗技术不断变化，进化出了更具威胁性的变种，传统的恶意代码分类手段已经无法准确对此类恶意代码进行分类。

为破解传统恶意代码分类中出现的问题，基于深度学习的恶意代码分类方式逐渐引起了广泛的重视。相较于传统方法，使用深度学习的方法能从大量训练样本中学习数据的内在规律，自行挖掘数据内部更深层的依赖关系。因此基于深度学习的恶意代码分类方法可以表现出更高的准确率[3]。

近年来，这一领域涌现出很多有意义的研究成果。文献[4]提出了一种多尺度特征融合卷积神经网络，利用深度学习实现基于可视化的恶意软件有效分类，可以防御恶意软件变体和混淆恶意软件。文献[5]提出了一种半监督方法，该方法集成了深度学习、特征工程、图像转换和处理技术，用于混淆恶意软件检测，准确率明显优于其它方法。文献[6]提出了一种基于静态特征的恶意软件分类算法（malware classification with SimHash and CNN，MCSC），该算法将反汇编后的恶意代码转换为基于 SimHash 的灰度图像，将转换后的灰度图输入 CNN 中对恶意代码进行分类。无论样本是否均匀分布，MCSC 均能有效对恶意软件进行分类。文献[7]将半监督生成对抗网络与深度卷积学习网络相结合，构建半监督深度卷积生成对抗网络，对恶意代码进行识别与分类，取得良好效果。文献[8]提出了一个混合的恶意代码分类框架，结合了静态和动态 2 种恶意代码分析方法，其中静态恶意代码可执行文件和动态进程内存转储文件通过填充空间的曲线转换为图像，从中提取视觉特征进行恶意代码分类，取得良好效果。文献[9]通过使用 Word2Vec 预训练策略来获得更紧凑的具有更少维度的向量，从而可以使参数更少和恶意软件特征表示更强。

上述基于恶意代码可视化的方法能够实现恶意代码变种的分类，一定程度上解决了代码混淆问题。然而，以上文献使用的单一序列特征或单一图像特征，特征的多样性不足，特征提取能力有限。

为了更好地表示恶意代码，挖掘恶意代码内部的特征信息，提高恶意代码特征提取能力和分类准确率，本文提出一种基于 BiTCNSA 的恶意代码分类方法，主要工作如下：

1）利用 n-gram 方法提取 OpCode 操作码特征，并将恶意代码转换为灰度图，融合 OpCode 特征与图像特征以展现不同的细节特征，增加特征多样性。

2）基于时域卷积网络（temporal convolution network，TCN），构建双向时域卷积网络（bidirectional temporal convolution network，BiTCN）用于恶意代码分类，增强特征提取能力。

3）提出了使用自注意力机制来捕捉数据内部的依赖关系，自适应地为数据分配不同的注意力权重，以提高模型分类能力。

1 相关工作

1.1 恶意代码特征提取

恶意代码图像特征最早由 NATARAJ 等人[10]提出，主要思想是将恶意代码的二进制文件转换成灰度图，利用图像中的纹理特征，采用图像处理的方法对恶意代码进行处理。

Tony 等人[11]在 2004 年最先提出了基于 ByteCode 提取 n-gram 特征应用于恶意代码处理的想法。2008 年 MOSKOVITCH 等人[12]提出基于 OpCode 提取 n-gram 特征的方法，比基于 ByteCode 提取的 n-gram 特征更加有效。

本文基于以上研究，充分将恶意代码的图像特征与在 OpCode 中提取的 n-gram 特征相结合，以混合特征作为双向时域卷积网络（BiTCN）的特征，有效利用了恶意代码的不同尺度的特征信息。

1.2 时域卷积网络 TCN

传统的卷积神经网络由于缺乏抓取长距离依赖信息的能力，一般不用于处理时序问题。但最近有研究表明，特定的卷积神经网络结构也可以有效对时序数据进行处理，这就是时域卷积网络。

时域卷积网络（TCN），最早由 BAI 等人[13]于 2018 年提出的，本质上是一种特殊的一维卷积。TCN 的网络结构在普通一维卷积的基础上添加了因果卷积和膨胀卷积，防止了信息从未来到现在的泄露，扩大了感受野，并使用了残差连接以增强模型的泛化性，在时间序列预测问题上拥有很好的表现。

相较于常用于时序问题的循环神经网络，TCN 可以对输入数据并行的处理，极大地提高了处理速度，在一些任务上甚至能超过循环神经网络相关模型。

本文在 TCN 的基础上，构建双向时域卷积网络用于恶意代码分类。

1.3　自注意力机制

注意力机制最早在视觉领域提出，Google Min 等人[14]指出了卷积神经网络的弊端，并提出了视觉注意力机制，使注意力机制逐渐引起关注。BAHDANAU 等人[15]提出将 Seq2Seq＋Attention 模型应用于机器翻译，首次将注意力机制应用在自然语言处理领域。随后自注意力和多头注意力等变体机制不断出现[16]。

自注意力机制作为注意力机制的一个变形，它不依赖其它的外部信息，只依靠自身的输入数据信息进行训练，就可以获得输入数据内部不同数据单元的注意力权重，以此捕获数据内部间的依赖关系，进而突出数据内更加重要的特征信息。

本文引入自注意力机制，充分利用注意力机制中的上下文学习能力，使每一个数据元对其它所有数据进行关注，深入挖掘数据间的依赖关系，从而提升恶意代码分类的准确度。

2　模型概述

本文设计的基于双向时域卷积网络（bidirectional temporal convolution network，BiTCN）与自注意力机制（Self Attention）的恶意代码分类模型（BiTCNSA），包括输入层、双向时域卷积网络层、自注意力层、softmax 层和输出层，其结构如图 1 所示。

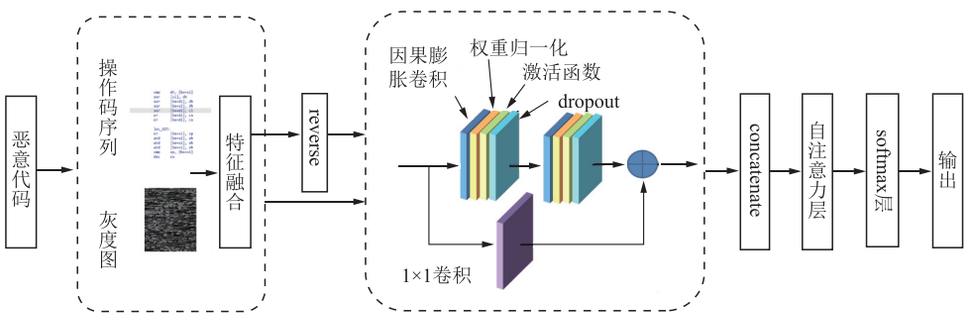


图 1　基于 BiTCNSA 的恶意代码分类模型

2.1　数据预处理

对数据处理的结果会直接影响到模型的最终结果，本文选择选取由微软公司提供的 Kaggle 数据集作为实验数据，分别通过恶意代码反汇编文件提取其序列特征和图像特征，采用特征融合方法将两者结合。

2.1.1　序列特征提取

首先，先从所有的恶意代码反汇编文件中分别提取操作码序列。其次对提取的操作码序列进行去重操作，以去除操作码序列中含有的大量冗余，避免耗费大量的计算时间。去重后的序列不仅保留了原有的序列信息，也更易于计算。再用 n-gram 方法提取去重后的操作码特征，在本文中，经过反复验证，选取 n-gram 的 $n=3$。经过反复验证，最后在具体的特征选择上选取每个分类出现次数高于的 500 的作为最终特征。特征提取流程如图 2 所示。

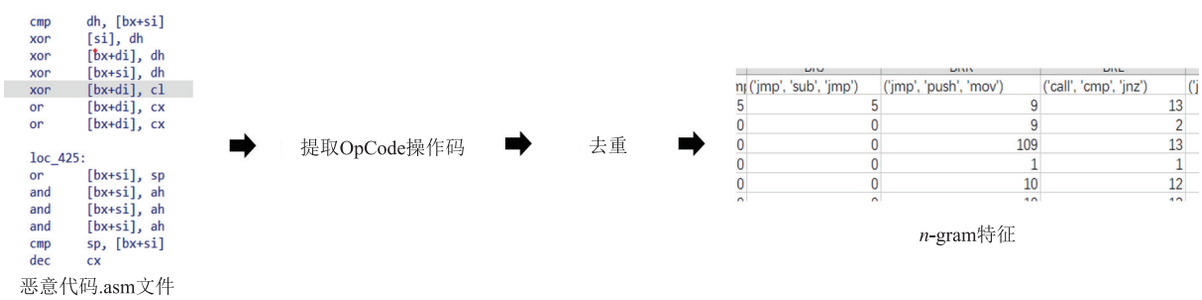


图 2　n-gram 特征提取

2.1.2　图像特征提取

将恶意代码反汇编文件转换成二进制流，从二进制数据中读入一个 8 位二进制数组成的向量，每个向量对应一个像素点，然后将向量的二进制值转换为十进制值，对应区间为[0，255]，其中 0 为黑色，255 为白色，即可将恶意代码转换为灰度图。由于恶意代码为一维数据，若将其转换为二维图像会影响原有的空间相关性，为了保留恶意代码原有的空间信息，因此不改变图像的维度，将恶意代码的转换为一维灰度图。图 3 为恶意代码生成灰度图过程。

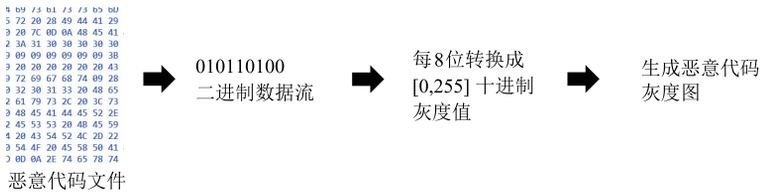


图 3 恶意代码生成灰度图流程

2.2 双向时域卷积网络

TCN 中的卷积网络具有可并行计算的特性，能够有效解决耗时过长的问题，已在多个领域证明比传统 RNN 甚至其相关变种更优。但是单一的 TCN 不能对从后到前的信息进行编码，导致无法学习当前特征项与后面特征项之间的关联。

为解决上述问题，本文构建在 TCN 的基础上，保留了因果膨胀卷积和残差连接，通过因果膨胀卷积学习数据的依赖关系，并使用残差连接消除有网络层数增加而导致的梯度消失问题。此外，一方面使用正向数据学习数据从前到后的依赖关系，另一方面通过逆向数据学习数据从后到前的依赖关系。最后通过将 2 个方向学习到的数据进行结合，从而获取正反两个传播方向间数据的依赖关系。

2.2.1 残差连接

在基于 BiTCNSA 的恶意代码分类模型中，整个双向时域卷积网络层由多个残差模块叠加而成，每个残差模块中包含了因果膨胀卷积，Weight Norm，激活函数 Relu 和 Dropout 以及一个 1×1 的卷积。

在每个残差块中，通过两因果膨胀卷积进行卷积操作。通过 Weight Norm 进行权重归一化，规范化隐含层的输入，解决梯度消失问题。Dropout 的加入能够有效解决模型过拟合问题。残差连接公式可表示为：

$$o=\mathrm{Activation}(x+F(x)) \tag{1}$$

式中：x 为输入；F 为残差网络。通过残差连接，能有效防止梯度消失，使神经网络更加稳定。残差连接结构如图 4 所示。

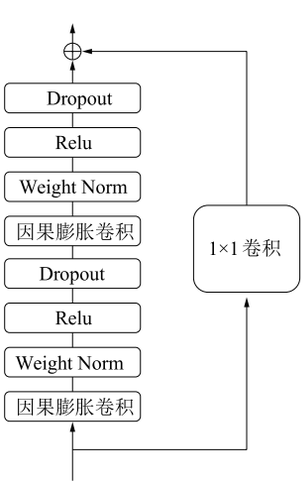


图 4 残差连接结构

2.2.2 因果卷积

为解决传统卷积神经网络对于时序数据处理时存在的信息从未来到现在的泄露问题，采用一种特殊的卷积神经网络结构，即因果卷积。

因果卷积是一种特殊的卷积神经网络，其原理是通过下一层 t 时刻的值和之前的值 $x_1,x_2,\cdots,x_t$ 来计算上一层 t 时刻的值 y_t，使得 y_t 接近于实际值。其公式为：

$$P(x)=P(x)=\prod_{t=1}^{T}P(x_t\mid x_1,x_2,\cdots,x_{t-1}) \tag{2}$$

对于当前时刻的值，因果卷积只用当前时刻之前的数据进行计算，同时限制不会对未来的数据有依赖，避免了信息的泄露，具有严格的约束性。

本文提出的基于 BiTCNSA 的恶意代码分类模型中的双向时域卷积网络层中，使用因果卷积，能有效解决时间序列数据中数据从未来对过去的泄露问题。因果膨胀卷积结构如图 5 所示。

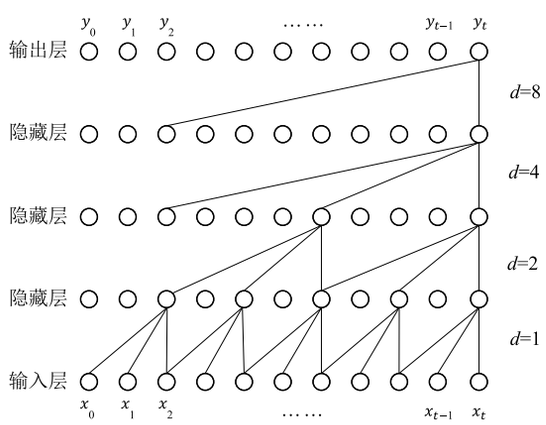


图 5 因果膨胀卷积

2.2.3 膨胀卷积

为解决传统卷积神经网络不善于抓取时序数据长距离依赖信息的问题，采用膨胀卷积。

膨胀卷积也叫空洞卷积，在传统卷积的基础上，膨胀卷积允许卷积时的输入数据间存在空格，即跳过部分输入使卷积核可以获得更大的感受野。具体来说，对于大小为 k 的卷积核，空洞数为 d，添加膨胀卷积的卷积核大小为 k'，其计算公式如下：

$$k'=k+(k-1)(d-1) \tag{3}$$

通常，随着网络层数的增加，d 的大小也要成指数型增长。如此一来，就能够实现将膨胀卷积网络感受野进行扩张。

本文模型的因果膨胀卷积共设置 4 层，每层的空洞数 d 分别设为 1、2、4、8，以更深模型层数、更多的空洞数来获取更高、更大的视野。

2.3　自注意力机制

本文引入自注意力机制以捕获数据间的依赖关系，挖掘深层次的特征关联性。首先将输入数据 I 与对应的权重矩阵相乘，转化得到查询向量 $\boldsymbol{Q}$，键向量 $\boldsymbol{K}$ 和值向量 $\boldsymbol{V}$，其中 $\boldsymbol{Q}=\boldsymbol{W}_q \cdot I$，$K=\boldsymbol{W}_k \cdot I$，$\boldsymbol{V}=\boldsymbol{W}_v \cdot I$。再计算查询向量和键向量间的相关性 $\boldsymbol{A}$，并进行 softmax 归一化得到 $\boldsymbol{A}'$，$\boldsymbol{A}'$表示每个查询向量分别对每个项输入数据的注意力权重，其公式为：

$$\boldsymbol{A}=\frac{\boldsymbol{Q}\cdot\boldsymbol{K}^{\mathrm{T}}}{\sqrt{\boldsymbol{d}_k}} \tag{4}$$

$$\boldsymbol{A}'=\mathrm{softmax}(\boldsymbol{A})=\frac{\exp(\boldsymbol{A})}{\sum_{k=1}^{n}\exp(\boldsymbol{A})} \tag{5}$$

最后，将值向量 $\boldsymbol{V}$ 与 $\boldsymbol{A}'$相乘得到最终的输出 O。其公式为 $O=\boldsymbol{A}'\cdot\boldsymbol{V}$

完整的自注意力机制公式为：

$$\mathrm{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V})=\mathrm{softmax}\left(\frac{\boldsymbol{Q}\cdot\boldsymbol{K}^{\mathrm{T}}}{\sqrt{\boldsymbol{d}_k}}\right)\boldsymbol{V} \tag{6}$$

3　实验与分析

3.1　数据集与实验环境

本实验数据集选择的是微软于 2015 年在数据竞赛平台 Kaggle 公开的恶意代码数据集。该数据集中的恶意代码共有 10 868 个样本，分为 9 个恶意代码家族，原始数据包含.asm 类型具有汇编语言代码的数据的文件以及二进制格式文件，没有 PE 标头。每个恶意软件文件都有一 ID、一个唯一标识文件的 20 个字符哈希值和一个 Class。本文所有实验使用 70％的数据集进行训练，30％的数据集进行测试。数据集如表 1 所示。

表 1　Microsoft Malware Classification 数据集

家族名	数量	类型
Ramnit	1 541	蠕虫
Lollipop	2 478	广告植入
Kelihos_ver1	398	后门
Kelihos_ver3	2 942	后门
Vundo	475	木马病毒
Simda	42	后门
Tracur	751	木马下载器
Obfuscator.ACY	1 228	混淆恶意代码
Gatak	1 013	后门

BiTCNSA 模型及所做的所有实验均在 Kreas 环境下完成，具体实验环境如表 2 所示。

表 2　实验环境配置

实验环境	具体配置
操作系统	Windows 11
CPU	Intel(R) Core(TM) i58300H CPU @ 2.30GHz　2.30 GHz
内存	16 GB
硬盘	500 GB
显卡	NVIDIA GeForce GTX 1050 Ti
开发框架	Keras 2.9.0/TensorFlow 2.9.1
开发语言	Python 3.10

3.2　评价标准

实验评价选用了准确率 Accuracy、精确率 Precision、召回率 Recall 和 F1 值等 4 个指标。各项评价指标定义分别为：

$$\mathrm{Accuracy}=\frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}} \tag{7}$$

$$\mathrm{Precision}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}} \tag{8}$$

$$\mathrm{Recall}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{9}$$

$$\mathrm{F1}=2\times\frac{\mathrm{Precision}\times\mathrm{Recall}}{\mathrm{Precision}+\mathrm{Recal}} \tag{10}$$

式中：TP 表示对正样本的正确预测；FP 表示对正样本的错误预测；FN 表示对负样本的错误预测；TN 表示对负样本的正确预测。

3.3　实验结果分析

为了充分验证本文提出的基于 BiTCNSA 的恶意代码分类方法的有效性，现设置如下实验：

实验 1：基于 BiTCNSA 的恶意代码分类实验；

实验 2：单特征和多特征融合对比分析实验；

实验 3：n-gram 取值分析实验；

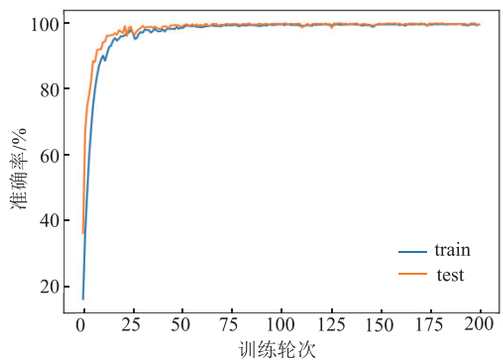
实验 4：本文模型与其它模型对比分析实验。

其中，实验 1 分析本文模型的分类能力；实验 2 验证多特征融合方法的有效性；实验 3 分析 n-gram 方法提取特征时，n 的取值的影响；实验 4 将本文模型与近年来其他模型进行对比，验证本文模型的有效性。

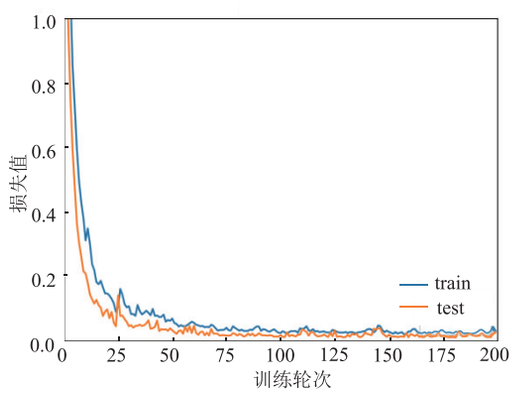
3.3.1　基于 BiTCNSA 的恶意代码分类实验(实验 1)

图 6 显示了模型训练过程中训练集和测试集的性能随训练批次的变化，其中图 6(a)是准确率随训练批次变化的曲线，图 6(b)是损失率随训练批次变化的曲线。黄线代表测试集，蓝线代表训练集。可以看到，模型能够快速收敛收敛。经过训练和测试，该模型的准确率达到 99.75％，损失率为 0.013 5。

为了清楚地观察模型的分类细节，绘制了模型的混淆矩阵，如图 7 所示。混淆矩阵中的主对角线的值表示恶意软件家族分类的真阳性率，而其他值表示恶意软件家族分类的假阴性率。可以看出该模型在多个家族分类效果优秀，仅在个别家族上分类存在误差。实验结果表明，该模型的准确率为 99.75%，精确率为 99.66%，召回率为 99.63%，F1 值为 99.69%。



(a)准确率



(b)损失率

图 6 模型训练信息

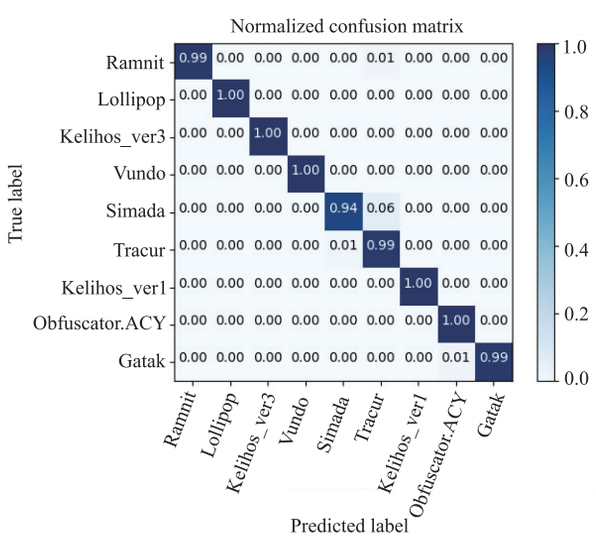


图 7 模型混淆矩阵

3.3.2 单特征和多特征融合对比分析实验(实验 2)

为了进一步提高对数据信息的提取能力，本模型在数据处理时分别用 n-gram 方法提取操作码序列特征和恶意代码的灰度图特征，并将两者进行融合。为验证该方法的有效性，设置对比实验，将操作码序列特征、灰度图特征以及混合特征进行对比，结果如表 3 所示。

从表中可以看出，混合特征的准确率、精确率、召回率和 F1 值等 4 项评价指标比单独操作码特征分别提高了 2.96%、2.41%、2.84%、2.81%，比单独图像特征分别提高了 12.22%、4.66%、30.49%、25.30%。

表 3 单特征和多特征融合对比分析

特征	评价指标/%			
	Accuracy	Precision	Recall	F1
操作码	96.79	97.25	96.79	96.88
图像特征	87.53	95.00	69.14	74.39
操作码＋图像特征	99.75	99.66	99.63	99.69

实验结果表明：操作码＋图像特征的混合特征的效果明显优于其中任何单个特征，对模型的效果提升明显，验证了方法的有效性。分析原因为：操作码序列特征和灰度图特征能够分别从不同尺度反映恶意代码的本质，将两者提取的特征相结合，能够丰富恶意代码的特征信息，产生互补作用，防止恶意代码混淆、加壳的影响，因此取得了更好的效果。

3.3.3 n-gram 取值分析实验(实验 3)

本模型数据在数据处理时用 n-gram 算法提取了恶意代码中操作码的特征，其中 n 的取值对模型效果有直接影响。为获取 n 最佳的取值，设置其余条件相同，将 $n=2$、3、4、5 等 4 种不同的取值结果进行对比，实验结果如图 8 所示。

从表中可以看出，相较于其他 n-gram 的取值，当 $n=3$ 时模型的准确率高达 99.75%，高于其它取值的准确率。当 n 取值高于 3 时，准确率逐渐降低。实验结果表明，$n=4$ 为 n-gram 的最佳取值。

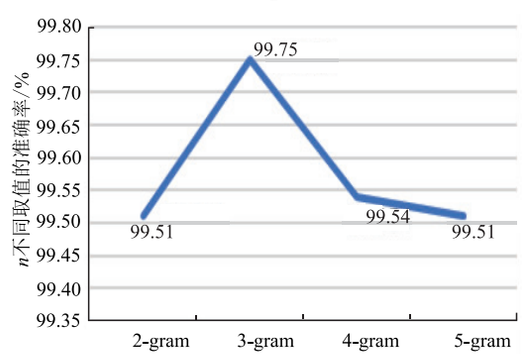


图 8 n-gram 不同取值对比

3.3.4 本文模型与其它模型对比分析实验

为进一步验证基于 BiTCNSA 恶意代码分类模

型的性能，现设置实验，将本模型与近年来其它恶意代码分类型进行对比，结果如表 4 所示。

从表 4 中可以看出，本文提出的基于 BiTCNSA 恶意代码分类模型准确率高达 99.75%，在各项评价指标上均优于其它所有方法。验证了本文方法的有效性。

表 4　不同模型实验结果对比

方法	特征	评价指标/%			
		Accuracy	F1-score	Precision	Recall
1DCNNIMIR[17]	灰度图	98.94			
RSGC[18]	操作码＋灰度图	98.90			
文献[19]	灰度图	97.50	94.00		
MCSC[20]	灰度图	98.86	98.07		
Orthrus[21]	字节＋操作码	99.24	98.72		
Word2VecTCN[9]	操作码＋API	97.50	97.50	97.60	97.50
本文模型	操作码＋灰度图	99.75	99.69	99.66	99.63

4　结语

本文提出了基于 BiTCNSA 的恶意代码分类方法，序列特征方面使用了 n-gram 提取 OpCode 操作码特征，图像方面使用一维图像而不是二维图像表示恶意代码特征，避免因为图像折叠带来的恶意代码图像特征中像素点之间不存在的局部相关性。将不同特征结合，从多角度利用恶意代码的特征信息。构建了双向时域卷积网络，使时域卷积网络能够充分利用前后 2 个方向的数据信息。引入自注意力机制进一步抓取数据内部的依赖关系。最后通过 softmax 层对恶意代码进行分类。实验结果表明，BiTCNSA 具有较高的准确率和收敛速度，验证了本文模型的可靠性，实现了提高准确率的恶意代码家族分类目标。

参考文献

[1] 国家互联网应急中心. 2022 年第 24 期网络安全信息与动态周报[EB/OL]. (2022-06-14)[2022-07-01]. https://www.cert.org.cn/publish/main/46/index.html.

[2] IRFAN A N, ARIFFIN A, MAHRIN M, et al. A Malware Detection Framework Based on Forensic and Unsupervised Machine Learning Methodologies[C]// 2020 9th International Conference on Software and Computer Applications. [S.l.]: ICSCA, 2020:194-200.

[3] 张杨，郝江波. 基于注意力机制和残差网络的恶意代码检测方法[J]. 计算机应用，2022，42(6)：1708-1715.

[4] WANG S, WANG J, SONG Y F, et al. Malicious Code Variant Identification Based on Multiscale Feature Fusion CNNs[J]. Computational Intelligence and Neuroscience, 2021, 2021(13):1.

[5] DAREM A, ABAWAJY J, MAKKAR A, et al. Visualization and Deep-Learning-Based Malware Variant Detection Using OpCode-Level Features[J]. Future Generation Computer Systems, 2021, 125:314-323.

[6] NI S, QIAN Q, ZHANG R. Malware Identification Using Visualization Images and Deep Learning[J]. Computers & Security, 2018, 77:871-885.

[7] 王栋，杨珂，玄佳兴，韩雨桐，等. 基于半监督生成对抗网络的恶意代码家族分类实现[J]. 计算机工程与科学，2022，44(5)：826-833.

[8] SHAO Y L, LU Y, WEI D, et al. Malicious Code Classification Method Based on Deep Residual Network and Hybrid Attention Mechanism for Edge Security[J]. Wireless Communications and Mobile Computing, 2022, 2022:3301718.

[9] SUN J, LUO X, GAO H, et al. Categorizing Malware via A Word2Vec-Based Temporal Convolutional Network Scheme[J]. Journal of Cloud Computing Advances Systems and Applications, 2020, 9(1): 1-14.

[10] NATARAJ L, KARTHIKEYAN S, JACOB G, et al. Malware Images: Visualization and Automatic Classification[C]//Proceedings of the 8th International Symposium on Visualization for Cyber Security. [S.l.]: ACM, 2011:1-7.

[11] ABOU-ASSALEH T, CERCONE N, KESELJ V, et al. N-Gram-Based Detection of New Malicious Code [C]// Proceedings of the 28th Annual International Computer Software and Applications Conference. Hongkong: IEEE Press, 2004.

[12] MOSKOVITCH R, FEHER C, TZACHAR N, et

al. Unknown Malcode Detection Using OPCODE Representation[C]//Proceedings of the 1st European Conference on Intelligence and Security Informatics. Esbjerg : ACM Press,2008:204-215.

[13] BAI S J , KOLTER J Z, KOLTUN V. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling[R]. ArXiv:1803.01271, 2018.

[14] MNIH V, GRAVES A, KAVUKCUOGLU K. Recurrent Models of Visual Attention.[J]. Advances in Neural Information Processing Systems, 2014, 3: 2204-2212.

[15] BAHDANAU D, CHO K, BENGIO Y. Neural Machine Translation by Jointly Learning to Align and Translate[R]. ArXiv:1409.0473,2014.

[16] VASWANI A , SHAZEER N , PARMAR N , et al. Attention Is All You Need[C]//Proceedings of the 31st International Conference on Neural Information Processing System.[S.l.], 2017:6000-6010.

[17] 王栋,杨珂,玄佳兴,等.基于一维卷积神经网络的恶意代码家族多分类方法研究[J].计算机应用与软件,2021,38(12):332-336,340.

[18] 陈小寒,魏书宁,覃正泽.基于深度学习可视化的恶意软件家族分类[J].计算机工程与应用,2021,57(22):131-138.

[19] GIBERT D, MATEU C, PLANES J, et al. Using Convolutional Neural Networks for Classification of Malware Represented as Images[J]. Journal of Computer Virology and Hacking Techniques, 2019, 15(1): 15-28.

[20] NI S, QIAN Q, ZHANG R. Malware Identification Using Visualization Images and Deep Learning[J]. Computers & Security, 2018, 77 : 871-885.

[21] GIBERT D , MATEU C , PLANES J . Orthrus: A Bimodal Learning Architecture for Malware Classification[C]// IEEE World Congress on Computational Intelligence (WCCI). Piscataway: IEEE Press, 2020.

(编辑:徐楠楠)