计算机科学
COMPUTER SCIENCE

# CheatKD:基于毒性神经元同化的知识蒸馏后门攻击方法

陈晋音, 李潇, 金海波, 陈若曦, 郑海斌, 李虎

引用本文

陈晋音, 李潇, 金海波, 陈若曦, 郑海斌, 李虎. CheatKD:基于毒性神经元同化的知识蒸馏后门攻击方法[J]. 计算机科学, 2024, 51(3): 351-359.

CHEN Jinyin, LI Xiao, JIN Haibo, CHEN Ruoxi, ZHENG Haibin, LI Hu. CheatKD:Knowledge Distillation Backdoor Attack Method Based on Poisoned Neuronal Assimilation [J]. Computer Science, 2024, 51(3): 351-359.

---

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.221200035

# CheatKD:基于毒性神经元同化的知识蒸馏后门攻击方法

陈晋音[1,2] 李 潇[1] 金海波[1] 陈若曦[1] 郑海斌[1,2] 李 虎[3]
1 浙江工业大学信息工程学院 杭州 310023
2 浙江工业大学网络空间安全研究院 杭州 310023
3 信息系统安全技术重点实验室 北京 100101
(chenjinyin@zjut.edu.cn)

**摘 要** 深度学习模型性能不断提升,但参数规模也越来越大,阻碍了其在边缘端设备的部署应用。为了解决这一问题,研究者提出了知识蒸馏(Knowledge Distillation,KD)技术,通过转移大型教师模型的"暗知识"快速生成高性能的小型学生模型,从而实现边缘端设备的轻量部署。然而,在实际场景中,许多教师模型是从公共平台下载的,缺乏必要的安全性审查,对知识蒸馏任务造成威胁。为此,我们首次提出针对特征KD的后门攻击方法CheatKD,其嵌入在教师模型中的后门,可以在KD过程中保留并转移至学生模型中,进而间接地使学生模型中毒。具体地,在训练教师模型的过程中,CheatKD初始化一个随机的触发器,并对其进行迭代优化,以控制教师模型中特定蒸馏层的部分神经元(即毒性神经元)的激活值,使其激活值趋于定值,以此实现毒性神经元同化操作,最终使教师模型中毒并携带后门。同时,该后门可以抵御知识蒸馏的过滤被传递到学生模型中。在4个数据集和6个模型组合的实验上,CheatKD取得了85%以上的平均攻击成功率,且对于多种蒸馏方法都具有较好的攻击泛用性。

**关键词**:后门攻击;深度学习;知识蒸馏;鲁棒性
**中图分类号** TP391

# CheatKD:Knowledge Distillation Backdoor Attack Method Based on Poisoned Neuronal Assimilation

CHEN Jinyin[1,2],LI Xiao[1],JIN Haibo[1],CHEN Ruoxi[1],ZHENG Haibin[1,2] and LI Hu[3]
1 College of Information Engineering,Zhejiang University of Technology,Hangzhou 310023,China
2 Institute of Cyberspace Security,Zhejiang University of Technology,Hangzhou 310023,China
3 Chinese National Key Laboratory of Science and Technology on Information System Security,Beijing 100101,China

**Abstract** With the continuous performance improvement of deep neural networks(DNNs),their parameter scale is also growing sharply,which hinders the deployment and application of DNNs on edge devices. To solve this problem,researchers propose knowledge distillation(KD). Small student models with high performance can be generated from KD,by learning the "dark knowledge" of large teacher models,realizing easy deployment of DNNs on edge devices. However,in the actual scenario,users often download large models from public model repositories,which lacks the guarantee of security. This may pose a severe threat to KD tasks. This paper proposes a backdoor attack for feature KD,named CheatKD,whose backdoor,embedded in the teacher model,can be retained and transferred to the student model during KD,and then indirectly poison the student model. Specifically,in the process of training the teacher model,CheatKD initializes a random trigger and optimizes it to control the activation values of some certain neurons of a particular distillation layer in the teacher model(i.e.,poisoned neuron),making their activation values fixed to enable poisoned neuronal assimilation. As the result,the teacher model is backdoored while this backdoor can resist to KD filtration and be transferred to the student model. Extensive experiment on four datasets and six model pairs have verified that CheatKD achieves an average attack success rate of 85.7%. Besides,it has good generality for various distillation methods.
**Keywords** Backdoor attack,Deep learning,Knowledge distillation,Robustness

到稿日期:2022-12-05 返修日期:2023-04-16
基金项目:国家自然科学基金(62072406);浙江省自然科学基金(DQ23F020001);信息系统安全技术重点实验室基金(61421110502)
This work was supported by the National Natural Science Foundation of China(62072406),Natural Science Foundation of Zhejiang Province,China(DQ23F020001) and Chinese National Key Laboratory of Science and Technology on Information System Security(61421110502).
通信作者:郑海斌(haibinzheng320@gmail.com)

## 1 引言

深度神经网络(Deep Neural Networks,DNNs)在图像分类[1-3]、目标检测[4-5]、语义分割[6]、自动驾驶[7-8]等领域取得了巨大的成功。性能良好的网络通常伴随着庞大的模型容量(即大量神经元和网络参数等),这会带来较高的计算和存储成本,极大地限制了模型在边缘端的部署应用。为了解决计算资源受限问题,研究人员提出了剪枝[9](Pruning)、量化[10](Quantization)和知识蒸馏[11](Knowledge Distillation,KD)等模型压缩方法。

作为一种主流的模型压缩方法,KD 使用大型教师模型指导小型学生模型训练的方式,传递教师模型中的"暗知识",使学生模型学习后能够具有媲美教师模型的良好性能[12-13]。目前,根据传播的知识载体不同,KD 的研究总体可以分为 3 类:基于预测向量的 KD(Logits-based Knowledge Distillation,LKD)、基于特征的 KD(Feature-based Knowledge Distillation,FKD)以及基于关系的 KD(Relation-based Knowledge Distillation,RKD)。其中,FKD 是最常用的知识蒸馏方法,它利用模型的中间层信息作为指导来训练学生模型,并取得了目前最佳的压缩性能。

在实际场景中,从头开始训练一个性能优秀的教师模型通常非常耗时,并且需要大量的计算资源和专业知识[14]。因此,从模型共享平台[1),2)]采用外包的预训练模型已成为用户获取性能优秀的教师模型的主流方式。在边缘端部署时,用户将对其进一步压缩,从而获得轻量、性能良好的学生模型。然而,由于模型共享平台未具备系统的安全性审查,且用户缺乏审查预训练模型的意识[15],模型的安全性无法得到保证,进而为在 KD 过程中攻击者操作公共教师模型来攻击用户的私人学生模型提供了机会,存在严重的安全风险。

现有对 KD 过程中潜在安全性风险的研究还处于起步阶段,Ge 等[16]首次提出了针对 LKD 的后门攻击 ADBA。他们引入了阴影模型来替代学生模型,进而模拟 KD 过程,并以此不断优化寻找能传递后门的触发器。但目前尚没有针对 FKD 的安全性风险的系统性研究。

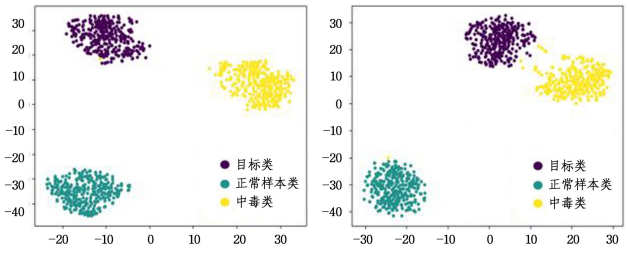
本文重点关注 FKD 的安全性问题,即后门攻击对 FKD 带来的潜在安全性风险。后门攻击旨在将预先定义的后门插入 DNN 中,并通过简单的外部后门触发器激活攻击,从而误导神经网络[17-19]。例如,Yao 等[20]提出了 Latent Backdoor Attack(LBA),他们将触发器关联中毒模型的内部层,从而在教师模型中安插后门,以便其在迁移学习过程中得以保留并转移到学生模型,进而实现对学生模型的操控。然而,有研究证明,已有的后门攻击方法无法通过 KD 过程使注入教师模型的后门保留并传递下来[21],因为在 KD 过程中只使用了干净样本。本文研究了 FKD 对后门攻击的鲁棒性,并且总结了目前后门攻击应用于 FKD 面临的挑战。

1)后门特征传递效率低。部分后门攻击方法通过拟合教师模型中毒样本与干净样本之间的特征,使学生模型学习到教师模型的决策边界,最终达到传递后门的效果。而这种方式由于特征和模型最终决策不具有强绑定关系,因此学生模型并不能完全习得中毒知识,导致后门传递效率低下。

2)教师-学生模型特征扭曲。在绝大部分情况下,教师模型和学生模型的中间层特征维度并不相同。为了消除这种不同,一般会采用 1×1 卷积层来对齐二者的维度。这就使得特征学习成为了一个间接过程,学生模型学到的"暗知识"会经过一定程度的扭曲。这种扭曲会影响教师模型中"后门暗知识"的传递。

为了深入研究后门在 FKD 中的传递现象,本文在 CIFAR-100[22]数据集上,使用 LBA 方法使教师模型 Wide residual networks 28-4[23](WRN28-4)中毒,使用 WRN16-4 作为学生模型进行后门的传递,最终通过 t-SNE[24]对学生模型的后门学习情况进行可视化分析。如图 1(a)所示,学生模型的目标类与中毒类之间的距离较远。一般情况下,攻击者将中毒类作为目标类训练,进而在模型内部插入后门。随着模型的不断训练,目标类与中毒类之间的距离也在不断缩小,最终导致模型将中毒类样本分类为目标类。因此,目标类与中毒类之间的距离,在一定程度上反映着学生模型学习后门的情况[25]。二者之间的距离较远,代表着此种后门在 FKD 的传递效率并不理想,进而直接影响到学生模型的后门植入。

为了解决上述问题,本文从神经元角度重新审视了 FKD 过程中的后门传递现象。具体地,我们将中毒样本输入至教师模型中,在固定层以激活值选取神经元,作为毒性神经元。为了使毒性神经元的输出特征尽可能鲁棒,我们以其激活值为触发器的优化目标,将其激活值趋于一个定值。通过学习特征,学生模型内部也会出现激活值稳定的毒性神经元,进而达到传递后门的目的。我们使用上述教师学生模型对和相同的 FKD 方法及设置进行实验,实验结果如图 1(b)所示。




(a)教师模型使用 LBA 植入后门 (b)教师模型使用本文方法植入后门

图 1 不同教师模型下学生模型的 TSNE 可视化

Fig. 1 TSNE visualization of student model with different teacher models

在相同尺度的坐标系下可以发现,图 1(b)中目标类与中毒类的距离明显比图 1(a)中的距离更近。这说明在后门的传递效率上,使用神经元来传递后门无疑是非常高效的。

基于上述实验结果,本文提出了一种基于毒性神经元同化的知识蒸馏后门攻击 CheatKD,其中,在公共教师模型中的后门可以在特征知识蒸馏过程中得以保留,从而转移到不同体系结构的学生模型中。首先初始化触发器,输入至教师

1) Jhttps://modelzoo.co

2) https://bigml.com

模型中，在固定层中选取激活平均值最大的神经元作为毒性神经元；其次，使用毒性神经元的激活值作为损失函数，不断优化触发器使其在毒性神经元上的输出趋于定值；最后，使用触发器生成中毒样本，对教师模型进行中毒操作。

本文的主要贡献如下：

1）探究了现存后门攻击无法有效攻击 FKD 过程中学生模型的问题，并总结了两个可能的原因：(1)后门特征传递效率低；(2)教师-学生模型特征扭曲。

2）首次提出了针对 FKD 的后门攻击方法 CheatKD。其中在教师模型中的后门可以在 FKD 中得以保留，并转移到不同体系结构的学生模型中。

3）在 4 个主流数据集上使用 4 种学生模型展开实验，验证了本文方法的有效性，并且使用多种不同类型的教师学生模型来验证泛用性。实验结果表明，本文方法在面对不同的数据集和异构网络时，仍具有良好的攻击性能；在面临不同数据和异构网络时，CheatKD 仍然具有良好的泛用性。

## 2 相关工作

本章简要介绍深度神经网络的知识蒸馏与后门攻击的相关工作。

### 2.1 知识蒸馏

目前对于 KD 的研究大致可以分为 3 类：基于预测向量的 KD、基于特征的 KD 以及基于关系的 KD。

基于预测向量的 KD[12,26-27]将教师模型的最后一层的输出向量作为指导，使学生模型直接模仿教师模型的最终预测。Hinton 等[12]使用温度因子来调整预测向量的平滑度。为了提升 LKD 的性能，Zhang 等[26]提出了一种相互学习的方式，以同时培养学生和教师。Mirzadeh 等[27]引入了一个名为“教师助理”的中等规模的模型来弥合教师和学生之间的差距。

基于关系的 KD[28-29]利用不同层或数据样本之间的关系来指导学生网络训练。Lee 等[28]使用径向基函数来分析特征之间的相关性，并使用奇异值分解来解决教师与学生特征之间的维度不匹配问题。Peng 等[29]提出了一种基于相关一致性的知识蒸馏方法，其中蒸馏的知识既包含实例级信息，又包含实例之间的相关性。使用关联一致性进行蒸馏，学生网络可以学习到实例之间的关联。

目前最先进的方法是基于特征的 KD，它可以利用教师模型的中间层信息作为指导来提升 KD 的性能[30-33]。Romero 等[30]首次从教师模型的中间层使用特征来指导学生的训练过程。Yim 等[31]提出了一种解决方案流程(FSP)，该流程由两层之间的 Gram 矩阵定义。FSP 矩阵总结了特征图之间的关系，它使用两层要素之间的内积来计算。Heo 等[32]认为不能只用神经元的激活值来做蒸馏约束，而应该使用神经元的激活区域来做约束。Heo 等[33]提出了一个新的特征蒸馏方法，该方法设计了新的蒸馏损失函数，可以使教师网络转移、学生网络转移、特征蒸馏位置以及距离函数协同作用。具体来说，蒸馏损失利用一个新设计的边缘 ReLU 的特征变换、一个新的特征蒸馏位置和一个部分 L2 距离函数来跳过冗余信息，防止对学生网络的压缩产生不利影响。

迄今为止，并没有任何 KD 方法关注其本身的安全性问题，例如是否会将教师模型带有的后门传递给学生模型。

### 2.2 后门攻击

后门攻击发生在模型训练阶段，攻击者将中毒样本注入训练数据集，从而在训练完成的深度学习模型中嵌入后门触发器，在测试阶段输入中毒样本，触发攻击。Gu 等[17]首次提出了后门攻击 BadNets，通过将中毒样本注入模型训练集中，成功注入了后门。Saha 等[34]提出了隐藏式后门攻击(Hidden Trigger Backdoor Attack，HTBA)。HTBA 使用一个隐蔽的触发器，并在特征空间上优化带有触发器的样本，使其中毒样本的特征尽可能接近目标类的样本。最终使得带有触发器的样本均被识别为目标类。

随着后门攻击的发展，其可迁移性越来越强。Li 等[19]提出了一种潜伏后门攻击(Latent Backdoor Attack，LBA)，使得中毒样本的特征与干净样本的特征尽可能接近。在训练教师模型中也使用特定的损失函数对模型进行训练，进而将触发器和模型中间层特征联系在一起。这种后门可以在迁移学习过程中得以保留并转移到学生模型中。Zhang 等[35]提出的后门攻击，通过对预训练模型添加一个简单的预训练任务，将触发器的输出表示限制为预定义的向量。此后门攻击可以忽略迁移前后的任务类型。Wang 等[36]在后门攻击过程中引入了知识蒸馏，提出了对抗知识蒸馏(Adversarial Knowledge Distillation，ADVKD)。ADVKD 通过知识蒸馏，减少了由标签翻转导致的模型异常特征，从而使后门模型能够绕过大多数防御。

现有后门攻击对知识蒸馏过程的影响性研究仍处于起步阶段。Ge 等[16]首次提出了针对 LKD 的后门攻击 ADBA。他们引入了阴影模型来替代学生模型，进而模拟知识蒸馏过程，并以此不断优化寻找能传递后门的触发器。然而，通过模拟知识蒸馏过程来优化触发器使得 ADBA 在时间和空间上的成本较大。不同于上述方法，CheatKD 不需要引入阴影模型来模拟知识蒸馏过程，这节省了大量的时间与空间成本。此外，对教师模型而言，由于知识蒸馏的实际作用机制未知，我们很难有效地利用学生模型的信息来优化教师模型的触发器。为了避免此类问题，CheatKD 只使用教师模型的内部信息来优化触发器，通过损失函数对毒性神经元进行同化操作，CheatKD 可以快速地迭代优化出相对应的触发器。

尽管有许多后门攻击可以在迁移学习中生效，但在使用知识蒸馏来迁移模型性能的情况下，现存的后门攻击并不能很好地将后门传递下去。

## 3 基于毒性神经元同化的知识蒸馏后门攻击方法

本章将从 CheatKD 的定义、步骤，如何选择中毒层及毒性神经元，如何选定鲁棒的特征这 3 个方面详述设计过程。

### 3.1 问题定义

本文将 CheatKD 定义为：给定一个教师模型 $F_t$，数据集 $D$，一个触发器 $P$ 和一个掩膜 $M$。对于给定的触发器 $P$ 和掩膜 $M$，我们将中毒样本定义如下：

$$x^* = (1-M) * x + M * P \tag{1}$$

使用最终优化后的 $P$ 和 $M$ 生成一系列中毒样本，与干净样本组成训练集，训练教师模型 $F_t$，使其变为带有后门的

教师模型$F_t^*$。我们的目标是，使得使用$F_t^*$作为教师模型训练的学生模型$F_s$也带有后门。

### 3.2 CheatKD 算法框架

为了更好地揭示 FKD 存在的后门安全性风险，本文提出了基于毒性神经元同化的知识蒸馏后门攻击方法 CheatKD，其算法框架如图 2 所示。首先生成随机的触发器 $P$ 和掩膜 $M$，进而生成一批中毒样本。随后，将中毒样本输入教师模型，在固定层中选取毒性神经元，如图 2 中红色神经元所示。然后，使用损失函数对触发器 $P$ 和掩膜 $M$ 进行迭代优化。值得注意的是，随着二者的不断更新，中毒样本也在不断更新，但毒性神经元不会随之更新。最后，使用优化完成的 $P$ 和 $M$ 生成最终的中毒样本，并联合干净样本一起训练教师模型。

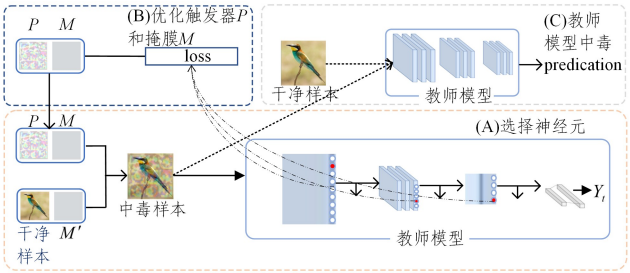



图 2 CheatKD 中毒过程(电子版为彩图)

Fig. 2 CheatKD poisoning process

#### 3.2.1 中毒位置和毒性神经元的选择

毒性神经元的输出包含大量的“后门暗知识”。为了提高教师模型特征中“后门暗知识”的含量，并且让学生模型尽可能学到“暗知识”，本文选择 FKD 效果最好的蒸馏层位置作为中毒层。

在 FitNet[29]中，蒸馏的位置是任意中间层的末端。然而，已有研究证实，这个位置并不是最佳位置[33]。目前性能最好的特征蒸馏位置是每一个图层组[22,26]的末端。图层组是具有相同空间大小的模型中间层的组合，其末端输出的特征更能代表一个模型所含的“暗知识”，这个位置被现有的特征蒸馏方法广泛使用[31,37]。基于此，本文选择图层组末端作为中毒层，并选择此层中一定数量的神经元作为毒性神经元，进而提升模型输出特征中“后门暗知识”的含量。在 CheatKD 中，为了选取毒性神经元，我们随机选择一批中毒样本，将其输入至教师模型中。将教师模型中毒层各神经元的平均激活值累加，取均值最大的一个神经元作为毒性神经元。

#### 3.2.2 毒性神经元同化

FKD 是一个间接的过程，“后门暗知识”在传递的过程中，由于特征整体的分布被扰乱，其变得无毒。本文提出了一种解决思路：使后门特征尽可能趋于一个值(即毒性神经元同化)。这样，即使后门特征的整体分布被扰乱，学生也可以在未被影响的后门特征中学到后门的整体特征。

在深度学习模型中，神经元的输出依据前一层神经元的输出与此神经元的权重矩阵相乘从而计算出来，并传递给下一层。给定输入 $X=\{x_1,x_2,\cdots,x_a\}$至目标模型中，设模型中毒层拥有 $a$ 个神经元，则第 $i$ 个样本神经元激活值矩阵 $\boldsymbol{N}^i$ 可以表示为：

$$\boldsymbol{N}^i=\begin{bmatrix} I_{1,1}^i & I_{2,1}^i & \cdots & I_{a,1}^i \\ I_{1,2}^i & I_{2,2}^i & \cdots & I_{a,2}^i \\ \cdots & \cdots & & \cdots \\ I_{1,b}^i & I_{2,b}^i & \cdots & I_{a,b}^i \end{bmatrix} \tag{2}$$

其中，$I$ 是神经元的激活值，$b$ 为中毒层的下层神经元总数。通过对激活值矩阵$\{\boldsymbol{N}^1,\boldsymbol{N}^2,\cdots,\boldsymbol{N}^n\}$按神经元取平均值，得到均值矩阵 $\boldsymbol{J}=\{I_1,I_2\cdots,I_a\}$，本文选取 $\delta$ 为矩阵 $\boldsymbol{J}$ 的最大值，最大值对应的神经元为毒性神经元。值得注意的是，在 CheatKD 中，当存在多个最大值对应的神经元时，默认选取索引值低的神经元为毒性神经元。最后，损失函数 $\mathcal{L}$ 的计算如下：

$$\mathcal{L}=\sum_{I\in N^*}(I-\delta)^2 \tag{3}$$

其中，$N^*$ 为选定的毒性神经元。本文在 4.4 节中对 $\delta$ 的取值进行了更多的探索。

#### 3.2.3 复杂度分析

1)时间复杂度：在 CheatKD 过程中，需要依次进行毒性神经元选取、触发器更新迭代以及教师模型中毒训练，因此以上 3 个步骤决定了 CheatKD 的时间复杂度。故 CheatKD 的时间复杂度为 $O(n)+O(z)+O(E)$，其中 $n$ 为深度模型中毒层神经元数目，$z$ 为更新触发器次数，$E$ 为模型训练迭代次数。

2)空间复杂度：算法的空间复杂度用来衡量所需的存储空间，即一个算法在运行过程中的临时占用存储空间。在 CheatKD 算法中，其空间复杂度与毒性神经元的个数、触发器更新次数以及模型训练次数相关。因此 CheatKD 的空间复杂度为 $O(n)+O(z)+O(E)$，其中 $n$ 为毒性神经元的个数，$z$ 为更新触发器次数，$E$ 为模型训练迭代次数。

## 4 实验结果与分析

本章首先介绍了实验设置，然后验证了本文提出的基于毒性神经元同化的知识蒸馏后门攻击方法的有效性，并且与主流的后门攻击 HTBA[34]，LBA[19]，BadNets[17]进行比较。

### 4.1 实验设置

#### 4.1.1 实验环境

所有实验均在一台搭载了 2×Intel(R) Xeon(R) Gold 5218R CPU @ 2.10 GHz、384 GB 系统内存和 8×NVIDIA A100 Tensor Core 40G GPU 的服务器上进行。集成开发环境为 Python3.6.0，采用深度学习框架 Torch 1.8.0。

#### 4.1.2 实验数据集

本文在 6 个教师模型、6 个学生模型上评估了 CheatKD 的性能，数据集包括 CIFAR-100 数据集、GTSRB 数据集、ImageNet-40 数据集、Flower-17 数据集。教师模型与学生模型的对应关系如表 1 所列。各数据集的详细信息如下：

1)CIFAR-100[22]：CIFAR-100 数据集有 100 个类，每个类包括 600 张 32×32 的彩色图像，每个类别有 500 个训练图像和 100 个测试图像。

2)GTSRB[38]：GTSRB 数据集包含各种环境中的 50 000 多张图像，其中 26 640 张用于训练，12 630 张用于测试。它从 43 个交通标志中提取出大小为 29×30 到 144×48 的彩色图像。

3）ImageNet-40[39]：ImageNet 数据集拥有 1 000 个分类，每个分类约有 1 000 张彩色图片。这些用于训练的图片总数约为 120 万张。测试集图片总数约为 10 万张。本文随机选取了 ImageNet 数据集中的 40 类作为实验数据集 ImageNet-40，其中每个类别有 1 300 个训练图像和 50 个测试图像。

4）Flower-17：Flower-17 数据集是一个细粒度的分类挑战，模型的任务是识别出 17 种不同种类的花。此图像数据集较小，每种花只有 80 张图片，其中 70 张用于训练，10 张用于测试，共包含 1 360 张图片。

表 1 教师模型-学生模型

Table 1 Teacher model-student model

| 模型设置 | 教师模型 | 学生模型 |
|---|---|---|
| a | WRN 28-4 | WRN 16-4 |
| b | WRN 28-4 | ResNet56 |
| c | Pyramid-200 | WRN 28-4 |
| d | Pyramid-200 | Pyramid-110 |
| e | ResNet152 | ResNet50 |
| f | ResNet50 | MobileNet |

#### 4.1.3 实验模型

本文在 CIFAR-100 数据集和 GTSRB 数据集上使用 WRN 28-4[23] 和 Pyramid-200[40] 作为教师模型，WRN 16-4 和 ResNet56[1] 作为 WRN 28-4 的学生模型，WRN 28-4 和 Pyramid-110 作为 Pyramid-200 的学生模型。在 ImageNet-40 和 Flower-17 数据集上，使用 ResNet152 与 ResNet50 教师学生对、ResNet50 与 MobileNet[41] 教师学生对来进行实验。

#### 4.1.4 对比算法

目前对 FKD 针对后门攻击的安全性研究较少，本文使用以下后门攻击方法作为 CheatKD 的对比算法。其中，BadNets 是最早提出的后门攻击方法，且在多模态领域已被证实具有良好的泛化性。与 CheatKD 控制的特征粒度类似，HTBA（Hidden Trigger Backdoor Attack）和 LBA（Latent Backdoor Attack）均从特征空间拟合角度安插后门；HTBA 拟合了中毒样本与干净样本的特征，其生成的中毒样本用肉眼无法察觉；LBA 将输出类别与模型的中间层表示联系在一起，其后门可以在迁移学习中得以保留。ADVKD 在后门攻击中引入了知识蒸馏思想，这可能会为后门在知识蒸馏中保留提供一定的帮助。各对比算法的详细信息如下：

1）HTBA[34]：使用一个隐蔽的触发器，并在特征空间上优化带有触发器的样本，使中毒样本的特征尽可能接近目标类的样本，最终使得带有触发器的样本均被识别为目标类。我们使用 HTBA 和位置优化的 HTBA 作为 CheatKD 的对比算法。为了更好地在 FKD 中传递“后门暗知识”，我们在 CheatKD 中，将带有触发器的样本在图层组末端输出的特征尽可能接近目标类样本。

2）BadNets[17]：BadNets 通过在样本的随机位置添加特殊的贴纸，进而生成中毒样本，并打上目标类标签；随后将中毒样本与干净样本混合变为模型中毒训练集，在此数据集上训练的模型都将带有后门。

3）（LBA）[20]：在特征空间中，LBA 迭代优化触发器，使得中毒样本与干净样本的特征尽可能接近。同时，在训练教师模型中，也使用特定的损失函数对模型进行训练，进而将触发器和模型中间层特征联系在一起。本文在模型的图层组末端选择特征。

4）ADVKD[36]：ADVKD 在后门攻击中引入了知识蒸馏思想，其将当前全局模型作为教师模型，将本地模型作为学生模型。ADVKD 利用教师模型与本地数据集生成了带有教师特征的数据，学生模型通过学习此类数据来达到知识蒸馏的效果。实验证明，ADVKD 可以减少由标签翻转导致的模型异常特征，从而使生成的后门模型能够绕过大多数防御。本文针对教师模型进行 ADVKD 攻击，并在模型的图层组末端选择特征。在 ADVKD 中，我们使用后门教师模型对干净样本进行标注。最后，使用干净模型学习中毒样本和干净样本及其标注，进而达到攻击的效果。

#### 4.1.5 参数设置和评估指标

在后门生成过程中，我们使用第 3 章介绍的方法制作了教师模型和触发器，并使用文献[33]中的方法作为 FKD，最后使用模型在测试数据集上的准确率和攻击成功率来评价 CheatKD。参数选择上，我们选择学习率为 0.001，毒性神经元每个中毒层 1 个，周期数 300 轮。在 4.3 节之后的实验中，若无特殊说明，默认分别使用 WRN 28-4 与 WRN 16-4 作为教师模型与学生模型，在 CIFAR-100 数据集上进行实验。

为了评估 CheatKD 的性能，本文使用学生模型在测试数据集上的识别准确率（Accuracy，ACC）和攻击成功率（Attack Success Ratio，ASR）作为评价指标。ACC 指标表示为：

$$ACC=\frac{TP+TN}{P+N} \tag{4}$$

其中，$TP$ 为真阳性样本数量，$TN$ 为真阴性样本数量，$P+N$ 为样本总数量。

ASR 可以表示为：

$$ASR=\frac{N_{\text{true}}}{N_{\text{total}}} \tag{5}$$

其中，$N_{\text{true}}$ 为预测正确的中毒样本数量，$N_{\text{total}}$ 为中毒样本的总数量。

### 4.2 实验结果

#### 4.2.1 CheatKD 的有效性分析

为了证明 CheatKD 的实际攻击性能，本文使用 CheatKD 在 4 个流行的数据集上使用 4 种教师学生模型对进行了对比实验。由于对比算法的攻击性能并不稳定，因此在表 2 中，我们选择对比算法数次实验的最大值来衡量此对比算法的攻击性能。对于 CheatKD，我们选择其数次实验的平均值来衡量其攻击性能。实验结果如表 2 所列。

从实验结果可以看出，CheatKD 的攻击性能远远优于其他算法，取得了目前最优的性能（State of the Art，SOTA）。具体而言，相较于对比算法，CheatKD 普遍取得了至少 20% 的性能提升，最高取得了 90%的性能提升。此外，HTBA 使用位置优化会显著提升攻击性能（普遍提升 20%左右）。即便如此，其攻击性能依旧不足以构成对知识蒸馏任务的威胁。相较于 CheatKD，HTBA 算法及其变体算法的学生模型在主任务准确率上有一定程度的下降，下降程度普遍在 3%左右，最大不超过 5%。BadNets 实验结果并不理想，整体的攻击

成功率不超过10%。LBA的攻击性能明显低于CheatKD，但却远远高于其他对比方法，这是因为LBA将触发器与模型的中间层表示联系在一起，进而保留了一定的“后门暗知识”。ADVKD性能优于BadNets，却低于其他对比算法。究其原因，可能是在建立后门的过程中，ADVKD使用干净模型学习样本的同时，也在不断拟合后门模型在此干净样本上的输出，这种模拟知识蒸馏的过程使得学习到的后门更易传播，进而影响了攻击成功率。

表2 CheatKD与对比算法在4个数据集上的攻击实验结果

Table 2 Attack experimental results of CheatKD and baselines on four datasets

| 数据集 | 模型设置 | 教师模型 | 不同方法下的学生模型性能 $ACC/\%/ASR/\%$ | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | $ACC/\%$ | CheatKD | HTBA | HTBA+位置优化 | BadNets | LBA | ADVKD |
| CIFAR-100 | a | 80.7±2 | 79.4/96.1 | 78.5/22.3 | 79.4/34.6 | 80.1/0.4 | 80.1/74.5 | 78.2/5.4 |
| | b | | 78.3/90.1 | 75.1/25.6 | 73.4/31.2 | 79.1/0.6 | 79.6/72.8 | 77.4/4.2 |
| | c | 85.5±1 | 83.1/99.9 | 82.6/24.5 | 80.6/34.3 | 83.5/0.9 | 84.5/76.4 | 83.2/3.3 |
| | d | | 83.2/100.0 | 81.6/23.6 | 81.4/30.1 | 84.3/1.0 | 82.4/77.8 | 82.8/3.9 |
| GTSRB | a | 98.1±2 | 100.0/92.6 | 98.5/16.9 | 95.6/40.5 | 99.5/0.6 | 99.5/74.1 | 99.7/6.4 |
| | b | | 100.0/93.5 | 96.5/24.6 | 98.2/38.1 | 99.1/0.5 | 100.0/75.4 | 99.4/2.8 |
| | c | 99.0±1 | 100.0/94.8 | 97.5/24.5 | 97.9/38.1 | 99.1/0.4 | 98.5/70.5 | 99.4/5.1 |
| | d | | 100.0/95.5 | 96.4/22.3 | 98.1/34.1 | 97.1/0.7 | 100.0/78.4 | 99.9/3.4 |
| ImageNet-40 | e | 80.2±2 | 84.2/53.3 | 82.5/13.4 | 83.1/28.6 | 82.5/0.5 | 83.4/49.0 | 83.4/0.9 |
| | f | 81.3±2 | 84.1/55.3 | 81.9/14.6 | 81.5/27.4 | 83.4/0.4 | 83.1/50.1 | 82.5/1.2 |
| Flower-17 | e | 99.0±1 | 90.4/75.8 | 82.4/14.2 | 85.9/25.4 | 89.9/0.9 | 90.9/70.4 | 89.4/1.0 |
| | f | 98.8±1 | 95.4/82.3 | 92.9/17.5 | 95.1/21.5 | 92.9/0.1 | 93.4/80.1 | 94.4/0.5 |

本文认为，CheatKD能达到SOTA性能是因为CheatKD将“后门暗知识”与毒性神经元强绑定，学生模型能够通过学习教师模型的特征生成毒性神经元，进而传递后门。为了进一步验证这一猜想，我们统计了教师模型与学生模型的毒性神经元。具体来讲，我们将一批中毒图片输入后门教师模型WRN 28-4中，并使用式(3)计算出教师模型固定层中各个神经元的损失值。结果显示，在教师模型的中毒层中，均是选中的毒性神经元的损失值最低。我们使用此教师模型帮助训练了一个学生模型，同样使用式(3)计算出学生模型中毒层中各个神经元的值。结果显示，在学生模型的图层组末端中，存在一个连续的神经元区间，此区间内的神经元的损失值最低。这充分说明CheatKD的后门可以通过FKD过程传递下来，并在学生模型上生成毒性神经元，即CheatKD的“后门暗知识”与毒性神经元是强绑定的。

从CheatKD各个数据集上实现的攻击效果来看，数据集尺寸会影响CheatKD性能。在CIFAR-100数据集上，CheatKD最高达到了100%的攻击成功率，平均ASR达到了96%，即使是最低的攻击效果，也达到了90%以上的攻击成功率。在GTSRB数据集上，CheatKD的攻击性能并无太大的波动，其最高ASR和最低ASR仅相差2%左右。总体而言，在小规模数据集CIFAR-100和GTSRB上，CheatKD表现出了强大的攻击性能。与小规模数据集上的表现不同，CheatKD在大规模数据集ImageNet-40和Flower-17上的平均ASR仅达到了66%。在ImageNet-40数据集上，CheatKD的攻击性能普遍不佳，其平均ASR仅达到了54%，最高ASR也仅有55%。与CheatKD在ImageNet-40数据集上的表现相比，其在Flower-17数据集上的表现有不小的提升，平均ASR达到了78%，最大ASR达到了82%。即使是最低的ASR(75.88%)，也远远大于CheatKD在ImageNet-40数据集上的最大ASR(55.35%)。

此外，CheatKD在大规模数据集中表现不佳，我们猜测原因可能是，不同尺寸的图片大小所包含的“暗知识”会随着图片尺寸的增大而变得复杂且多样，这就会导致学生在学习“暗知识”的同时，给予“后门暗知识”更少的关注，并最终导致CheatKD在学生模型上的攻击成功率不佳。

为了进一步验证此猜想，我们在ImageNet-40数据集上，以数据集所拥有的种类数量作为参数，对CheatKD进行了详细的实验。为了能更好地反映出CheatKD的攻击性能随数据集种类数的变化关系，我们取所有实验数据的平均值作为最终结果，实验结果如图3所示。随着数据集种类数目的增加，CheatKD的攻击性能缓慢降低。可以预见，当种类数量达到一定的数目时，CheatKD将丧失对知识蒸馏任务的威胁。

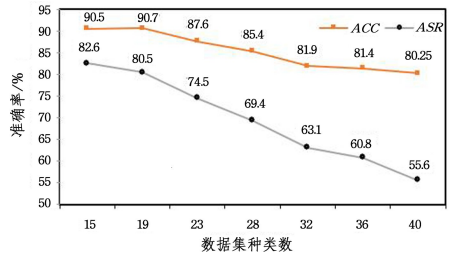



图3 大规模数据集种类数对CheatKD的影响

Fig. 3 Effect of large dataset classes on CheatKD

### 4.2.2 CheatKD的泛用性

本小节使用不同的教师模型-学生模型对和不同的FKD方法来说明CheatKD的泛用性。下面分别介绍不同的模型结构和FKD方法对CheatKD的影响分析。

1) 教师模型-学生模型结构差异实验

为了验证教师学生模型差异对CheatKD的影响，我们使用了4种不同类型的教师学生对，共计6对教师模型对，针对CheatKD做了详细的实验，实验结果如表3所列。在所有类型的学生模型中，相比教师模型，只改变深度的学生模型更容易受到后门的攻击，原因是其与教师模型结构的相似性过高，二者之间的特征维度相同，学生模型的学习更高效。相比之下，改变了通道数的学生模型效果稍差一些。CheatKD攻击性能最低的模型，是通道和深度均改变的学生模型。但即使是最低的攻击性能，也达到了90%的ASR。

在结构不同的教师学生模型对上，其学生模型的ASR要

比改变深度和通道的学生模型高。总体而言，无论对于什么类型的学生模型，CheatKD 都具有良好的攻击性能。本文还将 CheatKD 应用到迁移学习的场景中，将使用 CIFAR-100 数据集训练的后门模型迁移至 GTSRB 数据集中。其后门 ASR 依旧保留至 65.08%，以此模型作为教师模型，其学生模型也依旧保留 62.02%的 ASR。

表 3 CheatKD 在不同教师学生对上的攻击表现

Table 3 CheatKD's attack performance on different teacher-student pairs

(%)

| 压缩类型 | 教师模型 | *ACC* | *ASR* | 学生模型 | *ACC* | *ASR* |
|---|---|---|---|---|---|---|
| 深度 | WRN 28-4 | 80.1 | 99.2 | WRN 16-4 | 79.7 | 96.7 |
| 通道数 | WRN 28-4 | | | WRN 28-2 | | 92.9 |
| 深度和通道数 | WRN 28-4 | | | WRN 16-2 | | 90.1 |
| 结构 | WRN 28-4 | | | ResNet 56 | | 93.5 |
| 结构 | Pyramid-200 | 85.5 | 99.6 | WRN 28-4 | 83.4 | 95.6 |
| 结构 | Pyramid-200 | | | Pyramid-110 | | 92.3 |

2) FKD 差异实验

我们还使用了不同的 FKD 方法对 CheatKD 进行测试。为了充分体现各 FKD 方法针对后门攻击的鲁棒性，表 4 中的 ASR 数据均为 3 次实验的平均值。

表 4 现存 FKD 方法对后门攻击的鲁棒性

Table 4 Robustness of existing FKD methods to backdoor attacks

(%)

| FKD | *ACC* | *ASR* |
|---|---|---|
| AT[17] | 78.6 | 94.5 |
| COOFD[35] | 79.4 | 96.1 |
| FSP[30] | 77.6 | 90.7 |
| AB[38] | 79.0 | 98.5 |
| 平均值 | 78.6 | 94.9 |

随着 FKD 方法的不同，其学生模型的 ASR 虽有起伏，但全部超过了 90%，平均 ASR 达到了 94.95%。但从所有 FKD 方法的实验结果来看，虽然现存的 FKD 方法可以保存教师模型的主任务性能，但在面对鲁棒的后门攻击时却表现不佳。所有 FKD 方法的学生模型，其 ASR 均在 90%以上。

4.2.3 消融实验

本小节进行了消融实验，以衡量各个部分的实际效果。Baseline 是 BadNets，结果如表 5 所列。可以看出，使用特定的位置优化（3.2.1 小节）获得了最大的改进（提升超过 20%）。其次，使用特定的损失函数优化触发器（3.2.2 小节，式(3)）也获得了一定的改进（提升超过 15%）。虽然二者均对后门攻击鲁棒性有一定的提升，但经过提升的后门攻击仍不构成对知识蒸馏任务的威胁。总而言之，单个优化对于后门攻击鲁棒性的提升是有限的，结合所有提出的改进部分才最终使得 CheatKD 攻击性能显著提高。

表 5 消融实验

Table 5 Ablation experiment

| | Baseline | +位置优化 | +loss |
|---|---|---|---|
| ASR | 0.3 | 20.4 | 15.4 |
| 差异 | — | +20.1 | +15.1 |

4.2.4 参数敏感性实验

本小节通过改变 CheatKD 的超参数 $\delta$ 和毒性神经元的个数来对实验的影响进行分析，从而更加客观地对参数进行选择。

1) 超参数 $\delta$

首先对偏见神经元重训练的损失函数中的超参数 $\delta$ 进行了敏感性分析，超参数 $\delta$ 控制了毒性神经元最终趋向的值。我们在毒性神经元中的激活值分布中取了 5 个数据点。其中，max，min 和 mean 分别代表神经元激活值的最大值、最小值和均值。实验结果如表 6 所列。

表 6 超参数 $\delta$ 的敏感性

Table 6 Sensitivity of hyperparameter $\delta$

(%)

| $\delta$ | *ACC* | *ASR* |
|---|---|---|
| max | 41.3 | 10.8 |
| (max+min)/4 | 78.8 | 90.6 |
| mean | 79.4 | 96.1 |
| (max+min)/2 | 78.8 | 87.6 |
| min | 39.4 | 59.5 |

实验结果显示，$\delta$ 越趋近于均值，CheatKD 攻击效果越优秀；当 $\delta$ 的取值为均值时，CheatKD 取得了最佳的攻击效果；当 $\delta$ 的取值为最大值和最小值时，CheatKD 的攻击效果最差，且学生模型本身的性能也显著下降。究其原因，可能是一系列的极端数值导致干净样本与中毒样本在特征层面上的不兼容，并最终导致学生模型在 ACC 和 ASR 上的性能下降。

2) 毒性神经元个数

我们选择的毒性神经元的个数为 1 个（即 $n=1$），如表 1 所列，这又引起了一个问题：随着中毒神经元的增加，教师模型的特征所含有的“后门暗知识”也相应增多。为此，本文探究了模型的 ASR 与中毒神经元的数目的关系，实验结果如图 4 所示。

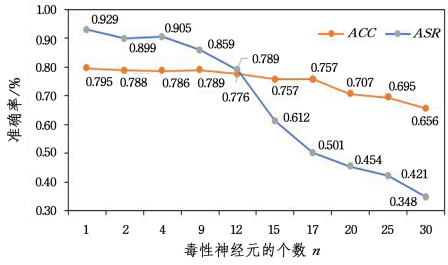



图 4 毒性神经元个数 $n$ 的敏感性

Fig. 4 Sensitivity of the number $n$ of toxic neuron

在一定程度内，随着毒性神经元数目 $n$ 的增加，学生模型的 ACC 与 ASR 并无明显变化。当 $n$ 取值在区间[1,9]内时，CheatKD 的攻击性能会有一定程度的起伏，但总体趋近于最佳攻击效果。当 $n$ 的取值超过 9 时，CheatKD 的攻击性能会

随着 $n$ 的增加而逐步降低，且对学生模型本身的性能也有负面影响。当 $n$ 的取值超过 25 时，学生模型的性能会明显下降。

#### 4.2.5 适应性防御

本节使用 NAD[42] 来防御 CheatKD。NAD 在知识蒸馏过程中引入了注意力机制，通过拟合干净教师模型与后门学生模型的注意力表示，成功地防御了绝大部分的后门攻击方法。在此实验中，我们选择在 CIFAR-10[21] 数据集上进行实验，实验结果如表 7 所列。

表 7 适应性防御对 CheatKD 的效果

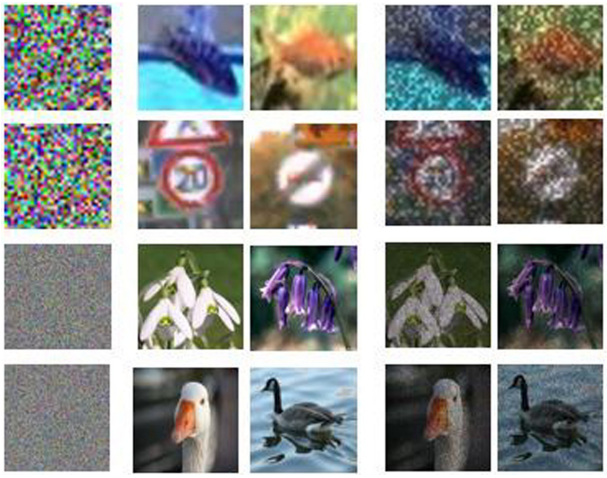
Table 7 Effects of adaptive defense on CheatKD

| 学生模型 | CheatKD | | NAD | |
|---|---|---|---|---|
| | *ACC* | *ASR* | *ACC* | *ASR* |
| WRN28-4 | 90.2 | 94.1 | 30.2 | 27.9 |
| WRN16-1 | 91.2 | 95.1 | 27.2 | 24.9 |
| ResNet56 | 92.5 | 94.5 | 31.4 | 30.4 |
| ResNet34 | 93.5 | 90.8 | 29.4 | 25.4 |

从整体的实验结果来看，NAD 可以较好地防御 CheatKD，但作为代价，模型的性能也随之骤降。从数值上看，模型的 ASR 与 ACC 下降的幅度近似，这或许是因为 CheatKD 所产生的后门足够鲁棒，致使 NAD 在抵御后门的同时，影响了模型的正常性能。

#### 4.2.6 触发器可视化

图 5 给出了 CheatKD 在实验中生成的触发器。触发器并不是事先设计好的，而是随着模型动态生成的。我们依次可视化了 CIFAR-100 数据集、GTSRB 数据集、Flower-17 数据集与 ImageNet-40 数据集中的触发器，并使用触发器生成了多张中毒样本。



(a)触发器 (b)干净样本 (c)中毒样本

图 5 CheatKD 对 4 个数据集生成的触发器及其有毒样本器示例

Fig. 5 Samples of triggers and toxic samplers generated by CheatKD on four datasets

**结束语** 本文提出了一种新的后门攻击 CheatKD。具体地，我们使用固定层中的特定神经元，通过将神经元内的激活值趋于一致化，建立了一个鲁棒且有效的后门，该后门隐藏在一个高性能的教师模型中，并且可以通过 KD 潜入其他学生模型中。我们在 4 个流行的数据集上验证了其有效性和实用性。此外，我们还通过攻击 6 种不同体系结构的学生模型来验证了模型的泛用性，使用迁移学习后的后门教师模型验证了其泛化能力。希望我们的工作可以引起人们对知识蒸馏技术的更多关注，并唤起人们对模型供应链的安全意识。然而，尽管 CheatKD 取得了 SOTA 的攻击性能，但未来仍需在以下方面进行优化改进：1)增加触发器的隐蔽程度；2)改进在多种类大规模数据集上的攻击效果。未来，我们将对后门在知识蒸馏中的可移植性进行明确的理论分析和解释。

## 参 考 文 献


[1] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 770-778.

[2] HU J, SHEN L, SUN G. Squeeze-and-excitation networks[C]// IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018: 7132-7141.

[3] MA N, ZHANG X, ZHENG H T, et al. Shufflenet v2: Practical guidelines for efficient cnn architecture design[C]// Proceedings of the European Conference on Computer Vision (ECCV). 2018: 116-131.

[4] HE K, GKIOXARI G, DOLLÁR P, et al. Mask r-cnn[C]// Proceedings of the IEEE International Conference on Computer Vision. 2017: 2961-2969.

[5] REN S, HE K, GIRSHICK R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[J]. arXiv: 1506.01497, 2015

[6] ZHAO H, SHI J, QI X, et al. Pyramid scene parsing network [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 2881-2890.

[7] CHEN J, BAI T. SAANet: Spatial adaptive alignment network for object detection in automatic driving[J]. Image and Vision Computing, 2020, 94: 103873.

[8] REN K, WANG Q, WANG C, et al. The security of autonomous driving: Threats, defenses, and future directions[J]. Proceedings of the IEEE, 2019, 108(2): 357-372.

[9] LI S, XU X Z. VGG16 optimization method based on double-angle parallel pruning[J]. Computer Science, 2021, 48(6): 227-233.

[10] SUN Y L, YE T Y. Convolutional neural network compression method based on pruning and quantization [J]. Computer Science, 2020, 47(8): 261-266.

[11] CHENG X M, DENG C H. Compression algorithm based on face recognition model based on label-free knowledge distillation[J]. Computer Science, 2022, 49(6): 245-253.

[12] HINTON G, VINYALS O, DEAN J. Distilling the knowledge in a neural network[J]. arXiv: 1503.02531, 2015.

[13] TANG W L, CENG J Y, HE W J. A lightweight model for orbital detection based on knowledge distillation[J]. Journal of Chongqing University of Technology (Natural Science), 2023, 37(9): 173-179.

[14] HUANG B, WANG Y H, ZHAO Y, et al. Intrusion Detection Model Incorporating MS-IRB and CAM[J]. Journal of Chinese Computer Systems, 2023, 44(7): 1586-1592.

[15] LIU Z, LI F, LI Z, et al. LoneNeuron: A Highly-Effective Fea-

ture-Domain Neural Trojan Using Invisible and Polymorphic Watermarks[C]// Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security. 2022：2129-2143.

[16] GE Y，WANG Q，ZHENG B，et al. Anti-distillation backdoor attacks：Backdoors can really survive in knowledge distillation [C]// Proceedings of the 29th ACM International Conference on Multimedia. 2021：826-834.

[17] GU T，DOLAN-GAVITT B，GARG S. Badnets：Identifying vulnerabilities in the machine learning model supply chain[J]. arXiv：1708. 06733，2017.

[18] LIU Y Q，MA S Q，AAFER Y，et al. Trojaning attack on neural networks[C]// Proceedings of the Network and Distributed System Security Symposium. 2017.

[19] LI X H，ZHENG H B，CHEN J Y，et al. Neural Path Poisoning Attack Method for Federated Learning[J]. Journal of Chinese Computer Systems，2023，44(7)：1578-1585.

[20] YAO Y，LI H，ZHENG H，et al. Latent backdoor attacks on deep neural networks[C]// Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security. 2019：2041-2055.

[21] YOSHIDA K，FUJINO T. Countermeasure against backdoor attack on neural networks utilizing knowledge distillation [J]. Journal of Signal Processing，2020，24(4)：141-144.

[22] KRIZHEVSKY A，HINTON G. Learning multiple layers of features from tiny images. Technical report[EB/OL]. (2018-04-08). https：//www. cs. toronto. edu/～ kriz/learning-features-2009-TR. pdf.

[23] ZAGORUYKO S，KOMODAKIS N. Wide residual networks [J]. arXiv：1605. 07146，2016.

[24] VAN DER MAATEN L，HINTON G. Visualizing data using t-SNE[J]. Journal of Machine Learning Research，2008，9(11)：2579-2605.

[25] WANG B，YAO Y，SHAN S，et al. Neural cleanse：Identifying and mitigating backdoor attacks in neural networks[C]// 2019 IEEE Symposium on Security and Privacy (SP). IEEE，2019：707-723.

[26] ZHANG Y，XIANG T，HOSPEDALES T M，et al. Deep mutual learning[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018：4320-4328.

[27] MIRZADEH S I，FARAJTABAR M，LI A，et al. Improved knowledge distillation via teacher assistant[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2020，34(4)：5191-5198.

[28] LEE S H，KIM D H，SONG B C. Self-supervised knowledge distillation using singular value decomposition[C]// Proceedings of the European Conference on Computer Vision (ECCV). 2018：335-350.

[29] PENG B，JIN X，LIU J，et al. Correlation congruence for knowledge distillation[C]// Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019：5007-5016.

[30] ROMERO A，BALLAS N，KAHOU S E，et al. Fitnets：Hints for thin deep nets[J]. arXiv：1412. 6550，2014.

[31] YIM J，JOO D，BAE J，et al. A gift from knowledge distillation：Fast optimization，network minimization and transfer learning [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017：4133-4141.

[32] HEO B，LEE M，YUN S，et al. Knowledge transfer via distillation of activation boundaries formed by hidden neurons[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2019：3779-3787.

[33] HEO B，KIM J，YUN S，et al. A comprehensive overhaul of feature distillation[C]// Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019：1921-1930.

[34] SAHA A，SUBRAMANYA A，PIRSIAVASH H. Hidden trigger backdoor attacks[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2020，34(7)：11957-11965.

[35] ZHANG Z，XIAO G，LI Y，et al. Red alarm for pre-trained models：Universal vulnerability to neuron-level backdoor attacks [J]. arXiv：2101. 06969，2021.

[36] WANG Y，FAN W，YANG K，et al. A Knowledge Distillation-Based Backdoor Attack in Federated Learning[J]. arXiv：2208. 06176，2022.

[37] SRINIVAS S，FLEURET F. Knowledge transfer with jacobian matching[C]// International Conference on Machine Learning. 2018：4723-4731.

[38] STALLKAMP J，SCHLIPSING M，SALMEN J，et al. Man vs. computer：benchmarking machine learning algorithms for traffic sign recognition[J]. Neural Network，2012；32：323-332.

[39] RUSSAKOVSKY O，DENG J，SU H，et al. Imagenet large scale visual recognition challenge[J]. International Journal of Computer Vision，2015，115(3)：211-252.

[40] HAN D，KIM J，KIM J. Deep pyramidal residual networks[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017：5927-5935.

[41] HOWARD A G，ZHU M，CHEN B，et al. Mobilenets：Efficient convolutional neural networks for mobile vision applications[J]. arXiv：1704. 04861，2017.

[42] LI Y，LYU X，KOREN N，et al. Neural attention distillation：Erasing backdoor triggers from deep neural networks[J]. arXiv：2101. 05930，2021.

**CHEN Jinyin**，born in 1982，Ph.D，professor. Her main research interests include artificial intelligence security，graph data mining and evolutionary computing.

**ZHENG Haibin**，born in 1995，Ph.D，lecturer. His main research interests include deep learning and artificial intelligence security.

（责任编辑：何杨）