计算机科学
COMPUTER SCIENCE

基于替代模型的批量零阶梯度符号算法

李炎达, 范纯龙, 滕一平, 于铠博

引用本文

李炎达, 范纯龙, 滕一平, 于铠博. 基于替代模型的批量零阶梯度符号算法[J]. 计算机科学, 2023, 50(11A): 230100036-6.

LI Yanda, FAN Chunlong, TENG Yiping, YU Kaibo. Batch Zeroth Order Gradient Symbol Method Based on Substitution Model [J]. Computer Science, 2023, 50(11A): 230100036-6.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.230100036

基于替代模型的批量零阶梯度符号算法

李炎达　范纯龙　滕一平　于铠博
沈阳航空航天大学计算机学院　沈阳 110136
(1147742111@qq.com)

摘　要　在面向神经网络的对抗攻击领域中,针对黑盒模型进行的通用攻击,如何生成导致多数样本输出错误的通用扰动是亟待解决的问题。然而,现有黑盒通用扰动生成算法的攻击效果不佳,且生成的扰动易被肉眼察觉。针对该问题,以典型卷积神经网络为研究对象,提出基于替代模型的批量零阶梯度符号算法。该算法通过对替代模型集合进行白盒攻击来初始化通用扰动,并在黑盒条件下查询目标模型,实现对通用扰动的稳定高效更新。在CIFAR-10和SVHN两个数据集上的实验结果表明,与基线算法对比,该算法攻击能力显著提升,其生成通用扰动的性能提高了近3倍。

关键词:卷积神经网络;通用扰动;对抗攻击;黑盒攻击;替代模型
中图法分类号　TP391

Batch Zeroth Order Gradient Symbol Method Based on Substitution Model

LI Yanda,FAN Chunlong,TENG Yiping and YU Kaibo
School of Computer Science,Shenyang Aerospace University,Shenyang 110136,China

Abstract　In the field of adversarial attacks for neural networks,for universal attacks on black-box model,how to generate universal perturbation which can cause most sample output errors is an urgent problem to be solved. However,existing black-box universal perturbation generation methods have poor attack effects and the generated perturbations are easy to be detected by the naked eye. To solve this problem,this paper takes the typical convolutional neural networks as the research object and proposed batch zeroth order gradient symbol method based on substitution model. This method initializes universal perturbation with white-box attacks on a set of alternative models,then realizes the stable and efficient updating of the universal perturbation by querying the target model under the black-box condition. Experimental results on two image retrieval datasets(CIFAR-10 and SVHN) show that the attack capability of this method is significantly improved,and the performance of generating universal perturbation is increased by 3 times.

Keywords　Convolutional neural network,Universal perturbation,Adversarial attack,Black-box attack,Substitution model

1　引言

近年来,神经网络研究发展迅速,其中深度神经网络在图像分类研究中的应用表现出良好的分类效果[1],其相关的安全问题也逐步受到重视。在研究中发现[2],如果在图像分类卷积神经网络的输入上添加一些微小的特定扰动,就会导致图像分类结果错误。这一过程称为对抗攻击,添加扰动后的神经网络输入称为对抗样本。对抗攻击可能使神经网络在实际应用中面临重大安全威胁,在例如医疗[3]、文本分类领域[4]等领域中造成难以估计的后果。因此,对抗攻击是神经网络在分类等任务研究中必须面对的关键问题,也是当前安全领域研究的热点内容,其对提高神经网络系统的安全性、鲁棒性[5]等有重要意义。

对抗攻击根据应用场景的不同可以分为不同类型,依据对模型的已知程度可以分为白盒攻击和黑盒攻击。白盒攻击[6]能够利用模型的结构、参数、梯度等信息开展对抗攻击,黑盒攻击[7]则只能查询神经网络的分类结果或分类概率进行对抗攻击。依据攻击样本数量的不同,可以分为单样本攻击和通用攻击,单样本攻击[6]生成的扰动只针对一个样本起攻击作用,通用攻击[8]则是利用一部分样本产生一个通用扰动,该扰动能使大部分其他样本的神经网络分类结果出现错误。

为了提升在黑盒条件下的神经网络通用攻击性能,本文提出的批量零阶梯度符号算法(Batch Zeroth Order gradient Symbol method,BZOS)利用查询到的神经网络分类概率计算近似梯度,经过多轮迭代计算通用扰动。为了进一步提升攻击效率,本文提出了基于替代模型的批量零阶梯度符号算法(Batch Zeroth Order gradient Symbol method based on Substitution model,BZOS-S)。本文的主要贡献如下:

1)提出了一种应用于黑盒场景的通用扰动生成算法(BZOS)。对该算法进行改进,使用替代模型进行扰动初始

基金项目:国家自然科学基金青年基金(61902260);辽宁省教育厅科学研究资助项目(JYT2020026)
This work was supported by the National Natural Science Foundation of China(61902260) and Scientific Research Project of Education Department of Liaoning Province(JYT2020026).
通信作者:范纯龙(FanCHL@sau.edu.cn)

化，并优化迭代策略，提出 BZOS-S 算法。在 2 个数据集和 3 个模型的多种组合上评估了二者的攻击性能。

2）BZOS 及 BZOS-S 算法能够在同等约束条件设置下，具有较高的攻击成功率，实现与白盒攻击方式接近的攻击效果，且远好于现有黑盒算法。

3）对 BZOS 及 BZOS-S 两种算法进行了对比，比较了替代模型攻击初始化的性能差异，以及两种算法的攻击效率。

本文第 2 节介绍了通用扰动生成领域在白盒与黑盒条件下的研究现状；第 3 节描述了本文提出的批量零阶梯度符号算法及其改进算法；第 4 节分别从通用扰动性能及算法攻击效率角度展开了实验，并分析了实验结果；最后总结全文并展望未来。

2 相关工作

近年来，对抗攻击领域已经有大量研究成果。在单样本白盒攻击方面，Goodfellow 等[6]提出了快速梯度符号法 FGSM，利用快速梯度符号生成扰动。BIM[9]和 PGD[5]均为 FGSM 的迭代方法，前者使用被攻击样本作为初始对抗样本进行迭代，后者则对被攻击样本添加随机扰动后再作为初始对抗样本。DeepFool[10]利用梯度优化损失函数进行迭代，以获取最小扰动，是目前白盒单样本扰动生成算法中最具代表性的高效算法之一。单样本黑盒攻击方面，文献[11]攻击算法利用扰动的可迁移性，通过训练与待攻击模型性能相似的替代模型，并对替代模型进行白盒攻击，来生成目标黑盒模型的扰动。这种方式训练的替代模型难以还原待攻击模型的分类边界，生成的对抗样本迁移性较低，并且训练时耗费大量计算资源。ZOO[12]在不利用替代模型的前提下，利用零阶优化生成扰动。OnePixel[13]通过差分进化生成仅改变个别像素点的扰动。通用攻击领域同样是研究热点，UAP[9]对样本集中的样本逐个利用 DeepFool 攻击，利用得到的单样本扰动进行通用扰动的迭代。CD-UAP[14]提出了一种利用不同的损失函数来处理目标类别和非目标类的图像，以生成通用扰动。UPSET[15]算法利用残差网络生成有目标的通用扰动。文献[16]则利用目标模型的梯度信息训练一个生成神经网络，该网络的输出结果即为通用扰动。HGAA[17]利用批量梯度上升，并通过超球面约束生成了通用对抗扰动，另外加入正则项用于约束扰动无穷范数，使扰动更难被察觉。黑盒通用攻击方面，文献[18]中提出了使用人工拼图进行无数据、无模型和无优化方法的通用攻击。DU-Attack[19]通过对角线分割进行降维，通过查询模型的迭代攻击方式，生成通用扰动。

目前，黑盒通用攻击方面的研究尚不全面，由于无法获取模型参数及梯度，算法的攻击性能较为一般。针对该问题，本文提出了一种基于零阶梯度上升符号法的通用扰动生成算法，通过计算扰动的近似梯度进行通用扰动迭代求解。为进一步提升算法效率，分别从通用扰动的初始化和迭代过程对该算法进行了改进，提出了基于批量零阶梯度符号法和替代模型的黑盒通用扰动生成算法。实验证明，本文提出的两种算法均能够生成具有较高的攻击成功率且不易被察觉的通用扰动，显著提升了黑盒通用攻击领域的算法性能。

3 黑盒对抗攻击算法

3.1 问题描述

通用对抗扰动是一个向量，它能够使样本集中大部分样本的神经网络分类结果发生变化。本文定义，$X\in[0,1]^d$ 表示一个 d 维空间上的样本集合；K 是一个图像分类网络，对每个样本 $\boldsymbol{x}\in X$ 输出一个分类标签 $K(\boldsymbol{x})$；寻找一个针对 X 的通用对抗扰动 $\boldsymbol{v}\in[-1,1]^d$，其性质及性能评价指标描述如下：

$$K(\boldsymbol{x})\neq K(\boldsymbol{x}+\boldsymbol{v}),\boldsymbol{x}\in X \tag{1}$$

$$h_{X,K}(v)=\frac{\sum_{\boldsymbol{x}\in X} sign(K(\boldsymbol{x})\neq K(\boldsymbol{x}+\boldsymbol{v}))}{|X|} \tag{2}$$

其中，$h_{X,K}(\boldsymbol{x})$表示针对分类网络 K 及数据集 X，该通用扰动的攻击成功率。通用对抗攻击的优化目标是寻找一个最优扰动$\boldsymbol{v}^*$，能够在指定分类网络 K 及数据集 X 上获得最大的攻击成功率，并满足 p-范数约束，使肉眼难以分辨样本的改变。

$$\boldsymbol{v}^*=\arg\max_{v}\{h_{X,K}(\boldsymbol{v})\mid\|\boldsymbol{v}\|_p\leqslant\varepsilon\} \tag{3}$$

3.2 批量梯度上升通用扰动求解算法

批量梯度上升是一种通用扰动求解算法，其原理类似神经网络通过梯度下降优化参数，当神经网络参数固定时，也可以将通用扰动作为参数进行优化，其优化目标是减小样本对应标签的输出。设 $\boldsymbol{v}$ 为样本 $\boldsymbol{x}$ 的对抗扰动，则$\boldsymbol{x}_{\text{adv}}=\boldsymbol{x}+\boldsymbol{v}$ 是其对抗样本；$p_k(\boldsymbol{x})$表示 $\boldsymbol{x}$ 在第 k 个分类上的概率，则 $K_x=\arg\max_k p_k(\boldsymbol{x})$为 $\boldsymbol{x}$ 的分类结果。扰动 $\boldsymbol{v}$ 使对抗样本的原标签分类概率 $p_{K(x)}(\boldsymbol{x}_{\text{adv}})<p_{K(x)}(\boldsymbol{x})$，$v$ 的更新方式为：

$$\boldsymbol{r}=\nabla_v loss(\boldsymbol{x}_{\text{adv}}) \tag{4}$$

$$\boldsymbol{v}=\boldsymbol{v}+\delta\boldsymbol{r} \tag{5}$$

其中，δ 为缩放因子，$\boldsymbol{r}$ 为扰动更新向量，$loss(\boldsymbol{x})$为损失函数，当 $loss(\boldsymbol{x})$值越大时，$p_{K(x)}(\boldsymbol{x}_{\text{adv}})$越小。对上述过程进行批量计算，每一次反向传播都能并行计算一批样本的梯度信息，计算出的扰动 $\boldsymbol{v}$ 能对该批样本中的大多数样本起到攻击作用，最终能够生成攻击成功率较高的通用扰动。

3.3 批量零阶梯度符号算法(BZOS)

本小节采用改进的批量梯度上升算法在黑盒条件下求解通用扰动 $\boldsymbol{v}$，并设置损失函数为 $loss(\boldsymbol{v})=\arg\max_{k\neq K(x)} p_k(\boldsymbol{x}_{\text{adv}})-p_{K(x)}(\boldsymbol{x}_{\text{adv}})$，其中$\arg\max_{k\neq K(x)} p_k(\boldsymbol{x}_{\text{adv}})$表示除正确分类外所有分类的最高概率值，损失函数大于 0 则代表攻击成功。由于黑盒场景下不能直接通过反向传播获取梯度，为实现黑盒下的批量梯度上升，本节采用零阶优化来计算近似梯度。对于损失函数 $loss(\boldsymbol{v})$，其梯度是一个向量，梯度在各个维度上的分量是损失函数在该维度上的偏导数，即：

$$\nabla_v loss(\boldsymbol{v})=\left[\frac{\partial loss(\boldsymbol{v})}{\partial \boldsymbol{v}_1},\frac{\partial loss(\boldsymbol{v})}{\partial \boldsymbol{v}_2},\cdots,\frac{\partial loss(\boldsymbol{v})}{\partial \boldsymbol{v}_d}\right] \tag{6}$$

因此计算近似梯度需要求出各维度的近似偏导数，我们通过计算对称差商获取近似偏导。

$$\frac{\partial loss(\boldsymbol{v})}{\partial \boldsymbol{v}_i}\approx\lim_{h\to 0}\frac{loss(\boldsymbol{x}_{adv}+h\boldsymbol{e}_i)-loss(\boldsymbol{x}_{\text{adv}}-h\boldsymbol{e}_i)}{2h} \tag{7}$$

其中，h 是一个非常小的常数(实验中取 0.0001)，$\boldsymbol{e}_i$ 是一个标准基向量，在第 i 个分量上取 1，这种方法计算得到的导数误差大约为 $O(h^2)$。该方法同样可以扩展，用于批量计算，此时能够获取该批样本的近似梯度平均作为扰动更新向量$\boldsymbol{r}^*$。

由于对称差商得到的$\boldsymbol{r}^*$仍然是不准确的,为了减小其误差带来的影响,本文取$\boldsymbol{r}^*$的符号$\boldsymbol{r}_s^* = \delta * sign(\boldsymbol{r}^*)$来更新通用扰动,步距严格可控。在通用扰动的迭代过程中,某些维度计算得到的近似偏导较小,但实际偏导值可能较大,该方法能够使这些维度获得稳定的更新。

最后,利用$\boldsymbol{r}_s^*$对通用扰动v进行更新,并添加了收敛系数η(实验中取0.99),实现动态更新率$\delta = \delta * \eta$,更新率随着轮次的增加逐步降低。经过多轮迭代后,该算法能够生成良好的通用对抗扰动,其具体步骤如算法1所示。

算法1 批量零阶梯度符号算法BZOS

输入:训练样本集合X;图像分类网络K;扰动范数约束ε;更新系数δ;收敛系数η;迭代轮次epochs;

输出:通用对抗扰动v

1. $\mathbf{v}=[0]^d$;X→{$batch_{x1}$,$batch_{x2}$,$batch_{x3}$,…,$batch_{xn}$} //初始化
2. for epoch in epochs do
3. for $batch_{\mathbf{x}}$ in {$batch_{x1}$,$batch_{x2}$,$batch_{x3}$,…,$batch_{xn}$} do
4. $\mathbf{x}_{adv}=\mathbf{x}+\mathbf{v}$ //生成对抗样本
5. for i in d do
6. $\mathbf{r}_i^* = \frac{loss(\mathbf{x}_{adv}+h\mathbf{e}_i)-loss(\mathbf{x}_{adv}-h\mathbf{e}_i)}{2h}$
7. $\mathbf{r}_i^* = \frac{loss(\mathbf{x}_{adv}+h\mathbf{e}_i)-loss(\mathbf{x}_{adv}-h\mathbf{e}_i)}{2h}$ //计算近似梯度
8. $\mathbf{r}_s^* = \delta * sign(\mathbf{r}^*)$ //取近似梯度符号
9. $\mathbf{v}=\mathbf{v}+\mathbf{r}_s^*$ //通用扰动更新
10. if $\|\mathbf{v}\|>\varepsilon$ do // 通用扰动模长约束
11. $\mathbf{v}=\varepsilon * \frac{\mathbf{v}}{\|\mathbf{v}\|}$ /
12. $\delta=\delta\times\eta$ //更新率衰减
13. return **v**

3.4 基于替代模型的批量零阶梯度符号算法(BZOS-S)

BZOS算法能够在黑盒场景下实现通用扰动的生成,但仍然需要大量查询模型输出结果,因此本文分别在通用扰动的初始化及迭代过程对其进行优化,以快速生成通用扰动。

3.4.1 基于替代模型的通用扰动初始化

BZOS算法的初始阶段扰动性能不佳,需要多轮迭代才能具备一定攻击能力,因此高效的通用扰动初始化是优化关键。因为通用对抗扰动具有一定的迁移性,所以本文使用白盒HGAA通用扰动攻击算法进行通用扰动初始化。该算法能够生成性能良好的通用扰动,将所得通用扰动迁移至其他模型上时,虽然仍具有一定攻击效果,但攻击性能有所下滑,尤其当迁移至不同结构的模型上时性能折损更加显著。为了增强通用扰动的迁移性,使其能对不同结构的模型同时起到攻击作用,本文围绕损失函数对HGAA算法进行了改进,使其具备对一个由多种不同结构分类器组成的替代模型集合$\{K_1', K_2', K_3', \cdots\}$的攻击能力。改进后的算法称为HGAA-M算法,改进后的损失函数为:

$$loss(\boldsymbol{v}) = \sum_{K^* \in [K_1', K_2', \cdots]} \min(loss_{org}(\boldsymbol{x}_{adv}), \tau)$$
$$\text{s.t. } \tau > loss_{org}(\boldsymbol{x}_0) \text{ while } \max_{k \neq K(x)} p_k(\boldsymbol{x}_0) \geqslant p_{K(x)}(\boldsymbol{x}_0) \tag{8}$$

其中,$loss_{org}$表示原HGAA算法中的损失函数,τ是一个常数,其值大于任一攻击成功对抗样本的损失值,即该对抗样本需满足其他分类概率最大值大于或等于标签分类概率。该损失函数的意义在于,当样本在某个模型上已经取得攻击效果,并且其损失值大于设定的阈值时,切断该样本在模型上的梯度,不进行反向传播,通用扰动的更新只由其他模型参与,使得最终所求通用扰动能同时对集合中所有替代模型起作用,具有较高的可迁移性。用该算法进行黑盒通用扰动求解的初始化,可以使通用对抗扰动具有更好的初始性能。

3.4.2 通用扰动迭代过程优化

为了减少模型的查询次数,提升扰动迭代效率,本文分别对近似梯度求解和扰动迭代方式进行了改进。

首先对近似梯度r^*的求解过程进行了优化,使用单侧差商拟合损失函数中的偏导数。

$$\frac{\partial loss(\boldsymbol{v})}{\partial \boldsymbol{v}_i} \approx \lim_{h \to 0} \frac{loss(\boldsymbol{x}_{adv}+h\boldsymbol{e}_i)-loss(\boldsymbol{x}_{adv})}{h} \tag{9}$$

此时计算一次扰动$\boldsymbol{v}$的各维度偏导时,式(9)中$loss(\boldsymbol{x}_{adv})$通过一次计算就能够获取,减少了一半查询次数。虽然该方法会导致误差增大至$O(h)$,但由于本文迭代时取近似梯度的符号,因此该误差并不会对算法的收敛趋势造成影响。

其次,本文迭代时改为仅计算部分维度近似梯度。每轮迭代时,在所有维度中随机抽取$d' \in (0, d)$个维度,仅计算扰动$\boldsymbol{v}$在这些维度上的偏导数,得到对抗样本的部分维度近似梯度,以更新通用扰动,可在不影响生成效果的前提下,将BZOS算法的每轮迭代分割为更多轮次的小计算开销迭代,使求解过程更细腻。

最后,实验表明,神经网络模型往往在靠近样本中间的维度上表现得更为敏感,损失函数$loss(\boldsymbol{v})$的偏导数在这些维度具有更大的绝对值,该部分实验将在4.3节详细阐述,因此,理论上通用扰动迭代时应在这些维度上有更大的更新率。本文设计了一个更新率权重矩阵$\boldsymbol{w}_\delta$,并将通用扰动更新的表达式修改为$\boldsymbol{v}=\boldsymbol{v}+\boldsymbol{w}_\delta \circ \boldsymbol{r}_s^*$,在每轮通用扰动迭代时同步计算该矩阵。具体地,在迭代过程中记录每轮迭代时各维度的近似梯度及迭代次数;筛选经过不少于2次近似梯度计算的维度,计算这些维度的近似梯度期望填充至期望掩码矩阵$\boldsymbol{mask}_{mean}$,并求出这些维度的期望的平均值$mask_e$;将$\boldsymbol{mask}_{mean}$中不为0的值减去$mask_e$,使$\boldsymbol{mask}_{mean}$的均值为0;最后,将$\boldsymbol{mask}_{mean}$进行放缩。

$$\boldsymbol{w}_\delta = \sigma * \frac{\boldsymbol{mask}_{mean}}{\max(\boldsymbol{mask}_{mean})-\min(\boldsymbol{mask}_{mean})}+1 \tag{10}$$

得到更新率权重矩阵$\boldsymbol{w}_\delta$,$\boldsymbol{w}_\delta$的均值始终为1,区间长度为σ。该计算$\boldsymbol{w}_\delta$的方法称为动态更新率(Dynamic Update Rate,DUR)方法,其能够使敏感区域更快速地进行迭代,从而提升扰动生成效率。

经过上述方法优化后,基于替代模型的批量零阶梯度符号算法如算法2所示。

算法2 基于替代模型的批量零阶梯度符号算法BZOS-S

输入:训练样本集合X;图像分类网络K;图像分类网络替代模型集合{K_1',K_2',K_3',…,K_n'};扰动范数约束ε;更新系数δ;收敛系数η;随机维度数量d';迭代轮次epochs

输出:通用对抗扰动v

1. $\mathbf{v}$=HGAA-M({K_1',K_2',…,K_n'},ε);X→{$batch_{x1}$,$batch_{x2}$,…,$batch_{xn}$} //初始化
2. for epoch in epochs do
3. for $batch_{\mathbf{x}}$ in {$batch_{x1}$,$batch_{x2}$,$batch_{x3}$,…,$batch_{xn}$} do
4. $\mathbf{x}_{adv}=\mathbf{x}+\mathbf{v}$ //生成对抗样本
5. pos={i_1,i_2,i_3,…,$i_{d'}$} s.t. i∈[1,d]//随机抽取d'个维度

6. for i in pos do
7. $\mathbf{r}_i^* = \frac{\text{loss}(\mathbf{x}_{adv}+h\mathbf{e}_i)-\text{loss}(\mathbf{x}_{adv}-h\mathbf{e}_i)}{2h}$
8. $\mathbf{r}_s^* = \delta * \text{sign}(\mathbf{r}^*)$
9. $\mathbf{w}_\delta = \text{DUR}(\mathbf{r}_s^*)$ //计算更新率权重矩阵
10. $\mathbf{v} = \mathbf{v} + \mathbf{w}_\delta \circ \mathbf{r}_s^*$ //通用扰动更新
11. if $\|\mathbf{v}\| > \varepsilon$ do // 通用扰动模长约束
12. $\mathbf{v} = \varepsilon * \frac{\mathbf{v}}{\|\mathbf{v}\|}$
13. $\delta = \delta \times \eta$ //更新率衰减
14. return $\mathbf{v}$

4 实验

在给定实验数据集和基准算法的条件下，本文对 BZOS 及 BZOS-S 两种黑盒通用扰动生成算法进行了验证。实验结果表明，两种算法均能够生成良好的通用扰动，性能较已有典型黑盒通用扰动生成算法显著提升，且改进后的 BZOS-S 算法具有更高的攻击效率。

4.1 实验条件设置

本文实验均在一台工作站上进行，其环境为：Windows 10、64 位操作系统，搭载 NVIDIA 2080Ti 显卡一块，CUDA 版本为 11.6，开发环境为 Python3.7 和 Pytorch1.10。

本文选用 CIFAR-10 图像分类数据集和 SVHN 门牌号码数据集作为实验用数据集，并基于这两个数据集的训练集，分别训练了 VGG[20]，NiN[21] 和 ResNet[22] 3 种不同规模的经典神经网络分类器。从各数据集中随机抽取 400 张样本作为通用扰动的训练集；从各数据集全部剩余样本中抽取分类器预测正确的样本，作为求得通用扰动的测试集。

本文选用 HGAA 白盒通用扰动生成算法与 DUA 黑盒通用扰动生成算法作为基线算法，与本文提出的 BZOS 算法及其改进算法 BZOS-S 进行性能对比，本文算法超参数设置如表 1 所列，基线算法均为默认参数。

表 1 各算法的实验参数

Table 1 Experimental parameters of each method

算法	参数
BZOS	$\delta=0.01, \eta=0.99, rounds=20, n=2$
BZOS_S	$\delta=0.01, \sigma=0.2, d'=512, \eta=0.99, rounds=60, n=2$

4.2 算法所得扰动性能对比

为了验证本文算法求解通用扰动的效率，在给定数据集上，使用 HGAA，DUA，BZOS，BZOS-S 这 4 种算法攻击目标模型，得到通用扰动。比较这些扰动的性能，选取 3 个指标将本文算法与基线算法对比，3 个指标分别为通用扰动的攻击成功率 $h_{X,K}(\boldsymbol{v})$、2-范数模长 $\|\boldsymbol{v}\|_2$ 以及无穷范数模长 $\|\boldsymbol{v}\|_\infty$。

首先，比较各组通用扰动的攻击成功率 $h_{X,K}(\boldsymbol{v})$。可以看到，本文提出的两种算法所得通用扰动在 CIFAR-10 和 SVHN 数据集上均表现良好，其攻击成功率接近甚至部分超过白盒 HGAA 算法所得扰动，也显著高于黑盒 DUA 算法所得扰动，其中，在 CIFAR-10 数据集上攻击成功率高出 DUA 算法约 135%，SVHN 数据集上高出 3 倍。本文的 BZOS 算法与 BZOS-S 算法相比，前者扰动的攻击成功率略高于后者，但算法其他性能方面后者更优，该部分结果在 4.2 节进行分析。表 2 的实验结果说明，本文算法在黑盒场景下能够生成攻击效果良好的通用扰动，其攻击性能远超现有黑盒算法，已经十分接近白盒算法的水平。

表 2 通用扰动生成算法性能对比

Table 2 Performance comparison of universal perturbation generation methods

攻击模型	数据集	HGAA($\|\boldsymbol{v}\|_2=2$)		DUA			BZOS($\|\boldsymbol{v}\|_2=2$)		BZOS-S($\|\boldsymbol{v}\|_2=2$)	
		$h_{X,K}(\boldsymbol{v})$	$\|\boldsymbol{v}\|_\infty$	$h_{X,K}(\boldsymbol{v})$	$\|\boldsymbol{v}\|_2$	$\|\boldsymbol{v}\|_\infty$	$h_{X,K}(\boldsymbol{v})$	$\|\boldsymbol{v}\|_\infty$	$h_{X,K}(\boldsymbol{v})$	$\|\boldsymbol{v}\|_\infty$
NiN	CIFAR-10	0.8059	0.212	0.6569	3.960	0.447	0.7988	0.144	0.7895	0.172
	SVHN	0.7341	0.163	0.2087	3.999	0.614	0.7646	0.155	0.7242	0.151
VGG	CIFAR-10	0.8577	0.199	0.6017	3.999	0.409	0.8529	0.153	0.8540	0.159
	SVHN	0.8239	0.147	0.2490	4.000	0.366	0.7822	0.168	0.7451	0.146
ResNet	CIFAR-10	0.8676	0.146	0.6145	4.000	0.304	0.8748	0.156	0.8608	0.169
	SVHN	0.7890	0.186	0.2121	4.000	0.319	0.7700	0.170	0.7472	0.150

其次，比较各组通用扰动的 2-范数模长 $\|\boldsymbol{v}\|_2$ 以及无穷范数模长 $\|\boldsymbol{v}\|_\infty$。其中，2-范数模长 $\|\boldsymbol{v}\|_2$ 是对通用扰动的整体约束；无穷范数模长 $\|\boldsymbol{v}\|_\infty$ 则反映了通用扰动在个别像素点上值的改变程度，无穷范数模长较大的通用扰动会存在肉眼可观察到的较为明显的像素点改变。二者数值越低，扰动质量就越高。本文 BZOS 算法、BZOS-S 算法所得扰动的 2-范数模长与基线中白盒 HGAA 算法相同，均为 2，低于黑盒 DUA 算法接近 50%；无穷范数方面，BZOS 算法、BZOS-S 算法所得扰动略低于 HGAA 算法，且显著低于 DUA 算法。实验结果表明，相比于现有的白盒及黑盒通用扰动生成方式，本文提出算法生成的扰动具有更高的质量，更不易被肉眼所察觉。

图 1 中实验结果表明，对初始分类不同的样本添加一样的通用扰动后，攻击成功的对抗样本均有一致的分类结果，本文推断出现该现象的原因是，通用扰动在有限的约束下表现为某种分类的分类特征。

此外，本文还比较了不同模长约束下，各算法的攻击成功率。4 种算法均选用 NiN 网络作为攻击目标，对 CIFAR-10 和 SVHN 两种数据集进行攻击，其中 DUA 在 2-范数模长为 2 时，在两个数据集上的攻击成功率分别为 0.251 和 0.028；2-范数模长为 3 时，攻击成功率分别为 0.452 和 0.131，相比另外 3 种算法性能较差，因此不在图中比较，其他算法性能曲线如图 2 所示。越靠近左上角，扰动模长越小，攻击成功率越高，其性能也就越好。实验结果表明，在不同模长约束下，本文算法相比黑盒 DUA 算法，均表现出巨大优势，且与白盒 HGAA 算法差距较小，在较小 2-范数模长约束下，攻击成功率甚至更高。

总体来说，在黑盒通用扰动生成领域，本文提出的两种算法均能够生成模长小、攻击成功率高的通用扰动，其性能及质量远优于现有算法，提升了约 5 倍，并达到了接近白盒通用

扰动生成算法的水平。

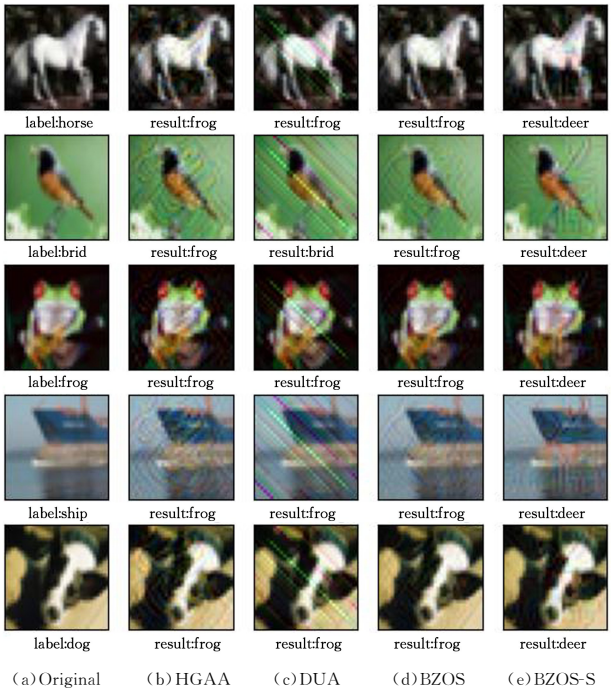


图 1　4 种通用扰动生成算法生成通用扰动图示

Fig. 1　Universal perturbation diagrams of four universal perturbation generation methods

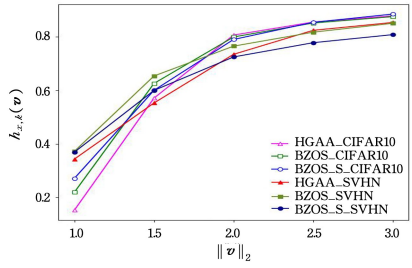


图 2　4 种通用扰动生成算法攻击成功率随 2-范数模长变化

Fig. 2　$h_{X,K}(\boldsymbol{v})$ varies with 2-norm modulus length among four universal perturbation generation methods

4.3　算法优化分析

本文提出的两种黑盒通用扰动生成算法中，BZOS-S 算法是对 BZOS 算法的优化。首先对优化过程进行分析，在 NiN 模型中使用 CIFAR-10 数据集绘制输入样本 x 的批平均梯度图像。如图 3 所示，在白盒条件下对 10 批样本进行梯度上升，对损失函数 $loss(\boldsymbol{x})$ 求梯度并取其绝对值。由于中间区域的维度绝对值更大，因此本文采用更新率权重矩阵更新扰动，使不同维度具有不同的更新步长的优化方式是有理论依据的。

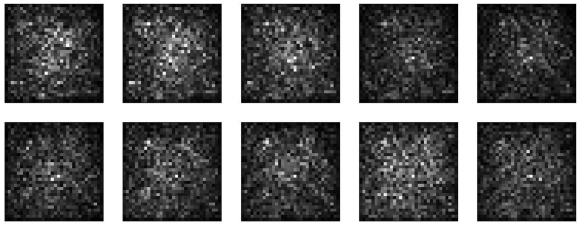


图 3　模型输入 x 的批平均损失函数梯度归一化图像

Fig. 3　Loss function gradient of model input image after normalization

其次，BZOS-S 算法中，使用了改进的 HGAA-M 算法攻击替代模型集合，该算法生成的初始通用扰动具有更强的迁移性。选用 HGAA 算法作为基线算法，攻击替代模型，比较两者生成的通用扰动在目标模型上的攻击成功率。实验中额外训练了 VGG，NiN 和 ResNet 这 3 种替代模型，HGAA-M 算法对替代模型的集合经过一次攻击生成通用扰动，HGAA 算法对 3 种替代模型分别进行攻击生成多个通用扰动，将上述扰动迁移至目标模型进行测试。表 3 中的实验结果证明了改进的 HGAA-M 算法在不同结构模型间的迁移性显著优于 HGAA 算法，在模型结构未知的黑盒攻击场景中适用范围更广，能够在不同结构的目标模型上起到更稳定的初始化作用。

表 3　HGAA 算法、HGAA-M 算法生成扰动迁移攻击成功率

Table 3　Migratory $h_{X,K}(\boldsymbol{v})$ of perturbations generated by HGAA and HGAA-M algorithms

目标模型	HGAA (VGG)	HGAA (NiN)	HGAA (ResNet)	HGAA-M
VGG	0.6947	0.4938	0.5898	**0.7575**
NiN	0.5327	**0.5737**	0.5303	0.5639
ResNet	0.6606	0.5278	**0.7516**	0.6700

最后对两种算法性能进行比较分析，本文选取 3 个指标，分别为算法生成通用扰动的攻击成功率 $h_{X,K}(\boldsymbol{v})$、算法查询次数 query_time 及算法耗时。其中，对于前两个指标，选用 NiN 模型及 CIFAR-10 数据集对算法进行实验，绘制 $h_{X,K}(\boldsymbol{v})$-query_time 图像用以逐轮分析，如图 4 所示。

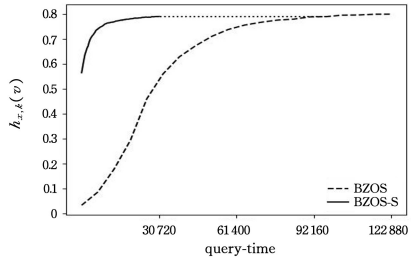


图 4　BZOS 算法、BZOS-S 算法攻击成功率随查询次数变化

Fig. 4　$h_{X,K}(\boldsymbol{v})$ varies with query_time between BZOS-S and BZOS-S algorithm

从实验结果中可以看出，两种算法均能稳定收敛，BZOS-S 算法仅需 30 720 次查询就能生成性能良好的通用扰动，相比 BZOS 算法减少了 1/4。图 4 中 BZOS-S 算法 $h_{X,K}(\boldsymbol{v})$-query_time 曲线起点攻击成功率更高，且具有更大的斜率，原因是 BZOS-S 算法使用了多替代模型攻击进行通用扰动初始化，改进后的迭代方式也具有更高的效率。实验中，BZOS-S 算法共迭代了 60 轮，每轮查询次数为 512 次；而 BZOS 算法共迭代 20 轮，每轮查询次数为 6 140 次。改进后的算法每轮查询次数更少，且可更灵活地设置算法迭代轮数，通用扰动的性能上升更加平滑，使算法能够灵活地应用在现实场景中。此外，如表 4 所列，改进后的 BZOS-S 算法耗时更短，仅需 BZOS 算法时间的 1/4。总的来说，本文所提 BZOS 算法所求扰动性能良好，并通过对其改进，在不降低通用扰动性能的前提下，大幅度提升了攻击效率。

表 4 BZOS 算法、BZOS-S 算法算法耗时

Table 4 Time consuming of BZOS and BZOS-S algorithms

(单位:s)

攻击模型及数据集		算法耗时	
		BZOS	BZOS-S
VGG	CIFAR-10	4858.54	1230.81
	SVHN	4894.25	1231.20
NiN	CIFAR-10	3727.35	971.63
	SVHN	3623.25	968.78
ResNet	CIFAR-10	4366.81	1177.66
	SVHN	4296.29	1160.10

结束语 针对黑盒场景下的图像分类神经网络的攻击问题,本文提出了一种批量零阶梯度符号算法,并在此基础上进行优化,提出了一种基于替代模型的批量零阶梯度符号算法。首先使用零阶梯度符号法解决了黑盒场景下通用扰动难生成问题;其次通过对通用扰动初始化问题及迭代过程进行优化,进一步提升了算法效率。两种算法均提供了明确的理论支撑,并通过实验验证了算法能够在生成扰动质量及生成效率方面有着明显的提升,性能显著优于现有算法。本文的研究内容是一个阶段性成果,在接下来的研究中,本文将尝试针对大型数据集进行攻击,进一步改进算法并开展实验,此外如何进一步降低算法计算开销也是值得优化的方向之一。

参 考 文 献

[1] DENG J, DONG W, SOCHER R, et al. Imagenet: A large-scale hierarchical image database[C]// 2009 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2009: 248-255.

[2] SZEGEDY C, ZAREMBA W, SUTSKEVER I, et al. Intriguing properties of neural networks [EB/OL]. https://doi.org/10.48550/arXiv.1312.6199.

[3] KOGA K, TAKEMOTO K. Simple Black-Box Universal Adversarial Attacks on Deep Neural Networks for Medical Image Classification [J]. Algorithms, 2022, 15(5): 144.

[4] HAO Z R, CHEN L, HUANG J C. Class Discriminative Universal Adversarial Attack for Text Classification[J]. Computer Science, 2022, 49(8): 323-329.

[5] MADRY A, MAKELOV A, SCHMIDT L, et al. Towards deep learning models resistant to adversarial attacks[EB/OL]. https://doi.org/10.48550/arXiv.1706.06083.

[6] GOODFELLOW I J, SHLENS J, SZEGEDY C. Explaining and harnessing adversarial examples[EB/OL]. https://doi.org/10.48550/arXiv.1412.6572.

[7] PAPERNOT N, MCDANIEL P, GOODFELLOW I, et al. Practical black-box attacks against machine learning[C]// Procee-dings of the 2017 ACM on Asia Conference on Computer and Communications Security. 2017: 506-519.

[8] MOOSAVI-DEZFOOLI S M, FAWZI A, FAWZI O, et al. Universal adversarial perturbations[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 1765-1773.

[9] KURAKIN A, GOODFELLOW I J, BENGIO S. Adversarial examples in the physical world[J]. arXiv: 1611.01236, 2016.

[10] MOOSAVI-DEZFOOLI S M, FAWZI A, FROSSARD P. Deepfool: a simple and accurate method to fool deep neural networks[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 2574-2582.

[11] ZHOU M, WU J, LIU Y, et al. Dast: Data-free substitute training for adversarial attacks[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 234-243.

[12] CHEN P Y, ZHANG H, SHARMA Y, et al. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models[C]// Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security. 2017: 15-26.

[13] SU J, VARGAS D V, SAKURAI K. One pixel attack for fooling deep neural networks[J]. IEEE Transactions on Evolutionary Computation, 2019, 23(5): 828-841.

[14] ZHANG C, BENZ P, IMTIAZ T, et al. Cd-uap: Class discriminative universal adversarial perturbation[C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2020: 6754-6761.

[15] SARKAR S, BANSAL A, MAHBUB U, et al. UPSET and ANGRI: Breaking high performance image classifiers[EB/OL]. https://doi.org/10.48550/arXiv.1707.01159.

[16] MOPURI K R, OJHA U, GARG U, et al. Nag: Network for adversary generation[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 742-751.

[17] FAN C L, LI Y D, XIA X F, et al. A general adversarial attack method based on random gradient Ascent and spherical projection[J]. Journal of Northeastern University: Natural Science, 2022, 43(2): 168-175.

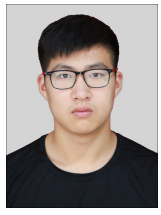
[18] ZHANG C, BENZ P, KARJAUV A, et al. Data-free universal adversarial perturbation and black-box attack[C]// Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 7868-7877.

[19] WU J, ZHOU M, LIU S, et al. Decision-based universal adversarial attack[EB/OL]. https://doi.org/10.48550/arXiv.2009.07024.

[20] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition [EB/OL]. https://doi.org/10.48550/arXiv.1409.1556.

[21] LIN M, CHEN Q, YAN S. Network in network [EB/OL]. https://doi.org/10.48550/arXiv.1312.4400.

[22] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 770-778.



LI Yanda, born in 1999, postgraduate. His main research interests include neural network counterattack and reinforcement learning.

FAN Chunlong, born in 1973, Ph.D, professor, postgraduate supervisor. His main research interests include interpretability of neural networks, complex network analysis and intelligent system verification.