软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software [doi: 10.13328/j.cnki.jos.007056]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于局部扰动的时间序列预测对抗攻击*

张耀元[1,2], 原继东[1,2], 刘海洋[2], 王志海[2], 赵培翔[2]

[1](交通大数据与人工智能教育部重点实验室 (北京交通大学), 北京 100044)

[2](北京交通大学 计算机与信息技术学院, 北京 100044)

通信作者: 原继东, E-mail: yuanjd@bjtu.edu.cn

摘　要: 时间序列预测模型已广泛应用于日常生活中的各个行业, 针对这些预测模型的对抗攻击关系到各行业数据的安全性. 目前, 时间序列的对抗攻击多在全局范围内进行大规模扰动, 导致对抗样本易被感知. 同时, 对抗攻击的效果会随着扰动幅度的降低而明显下降. 因此, 如何在生成不易察觉的对抗样本的同时保持较好的攻击效果, 是当前时间序列预测对抗攻击领域亟需解决的问题之一. 首先提出一种基于滑动窗口的局部扰动策略, 缩小对抗样本的扰动区间; 其次, 使用差分进化算法寻找最优攻击点位, 并结合分段函数分割扰动区间, 进一步降低扰动范围, 完成半白盒攻击. 和已有的对抗攻击方法在多个不同深度模型上的对比实验表明, 所提出的方法能够生成不易感知的对抗样本, 并有效改变模型的预测趋势, 在股票交易、电力消耗、太阳黑子观测和气温预测这 4 个具有挑战性的任务中均取得了较好的攻击效果.

关键词: 时间序列预测; 对抗攻击; 对抗样本; 半白盒攻击; 滑动窗口; 差分进化

中图法分类号: TP18

中文引用格式: 张耀元, 原继东, 刘海洋, 王志海, 赵培翔. 基于局部扰动的时间序列预测对抗攻击. 软件学报. http://www.jos.org.cn/1000-9825/7056.htm

英文引用格式: Zhang YY, Yuan JD, Liu HY, Wang ZH, Zhao PX. Adversarial Attack of Time Series Forecasting Based on Local Perturbations. Ruan Jian Xue Bao/Journal of Software (in Chinese). http://www.jos.org.cn/1000-9825/7056.htm

Adversarial Attack of Time Series Forecasting Based on Local Perturbations

ZHANG Yao-Yuan[1,2], YUAN Ji-Dong[1,2], LIU Hai-Yang[2], WANG Zhi-Hai[2], ZHAO Pei-Xiang[2]

[1](Key Laboratory of Big Data & Artificial Intelligence in Transportation (Beijing Jiaotong University), Ministry of Education, Beijing 100044, China)

[2](School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China)

Abstract: Time series forecasting models have been widely used in various domains of daily life, and the attack against these models is related to the security of data in applications. At present, adversarial attacks on time series mostly perform large-scale perturbation at the global level, which leads to the easy perception of adversarial samples. At the same time, the effectiveness of adversarial attacks decreases significantly with the magnitude shrinkage of the perturbation. Therefore, how to generate imperceptible adversarial samples while maintaining a competitive performance of attack is an urgent problem that needs to be solved in the current adversarial attack field of time series forecasting. This study first proposes a local perturbation strategy based on sliding windows to narrow the perturbation interval of the adversarial sample. Second, it employs the differential evolutionary algorithm to find the optimal attack points and combine the segmentation function to partition the perturbation interval to further reduce the perturbation range and complete the semi-white-box attack. The comparison experiments with existing adversarial attack methods on several different deep learning models show that the proposed method can generate less perceptible adversarial samples and effectively change the prediction trend of the model. The proposed method achieves sound attack results in four challenging tasks, namely stock trading, electricity consumption, sunspot observation, and temperature prediction.

* 基金项目: 中央高校基本科研业务费专项 (2022JBMC011); 国家自然科学基金 (61702030)
收稿时间: 2023-03-29; 修改时间: 2023-05-03, 2023-08-01; 采用时间: 2023-09-03; jos 在线出版时间: 2024-01-31

Key words: time series forecasting; adversarial attack; adversarial sample; semi-white-box attack; sliding window; differential evolution

时间序列通常代表一组按照时间排列的随机实值型变量[1], 它广泛存在于日常生活的各行各业. 随着深度神经网络 (deep neural networks, DNNs) 不断发展, 研究者尝试将其扩展到时间序列中, 如长短期记忆网络 (long short-term memory, LSTM)[2]和时序卷积神经网络 (temporal convolutional network, TCN)[3]等已成功应用于时间序列预测问题[4]. 在对抗样本概念被提出后, 研究人员发现深度学习中的多数模型易受到对抗样本的干扰从而作出不正确的判断, 这一行为被称为对抗攻击[5], 例如对于时间序列分类模型, 主要利用样本间的差异性进行分类[6], 而其差异性也是成功进行对抗攻击的关键点之一. 相关实验表明, DNNs 在时间序列分类任务中对攻击的鲁棒性较低, 导致其易受到攻击影响从而做出错误的分类[7].

迄今为止, 时间序列预测在多个行业的应用尤为重要, 然而相关的对抗攻击研究较少. 例如, 地方电网公司会通过智能系统根据历史数据预测用户在不同时间段的用电情况, 从而有效地在不同时间段供给适度的电力负荷. 这不仅能够保障用户的用电需求, 而且能够节省电力资源. 然而, 当输入的用户数据被攻击后, 可能会导致智能系统给出错误的预测从而造成电力负荷超载, 严重情况下会导致全地区停电[8]. 因此, 针对时间序列预测的对抗攻击研究是必要的.

目前, 对抗攻击多应用于图像识别和分类领域, 攻击方法主要通过改动图像中的一个或多个像素点信息来生成对抗样本[9]. 受到人眼所能察觉到的范围限制, 图像中的个别像素点变动是很难感知到的. 然而, 对于时间序列这种连续型且呈线条状的数据样本, 与图像恰好相反, 对抗攻击的难度在于如何生成人眼无法轻易发现区别的对抗样本. 现阶段, 已有的图像领域的白盒攻击方法主要是基于梯度实现的. 图 1 为具有代表性的快速梯度符号法 (fast gradient sign method, FGSM)[10]应用到时间序列预测中的示例, 黑色曲线为原始的输入样本, 灰色曲线为添加扰动后的对抗样本. 如图 1 所示, 依据梯度生成的扰动作用在全局序列上, 其不仅会因全局扰动幅度过大降低其隐匿性, 也可能因局部区域扰动强烈而易被感知. 例如黑色虚线框中为其中一段局部区间内原始样本和对抗样本的对比, 相比其他区间其扰动较大, 人眼可以轻易察觉样本变化. 为了解决这一问题, 一个直观的想法就是减少扰动区间和扰动幅度, 但是区间减少和幅度降低的同时会导致攻击效果下降. 因此, 在生成不易察觉的对抗样本的同时能够保证较好的攻击效果, 是本文的研究重点.

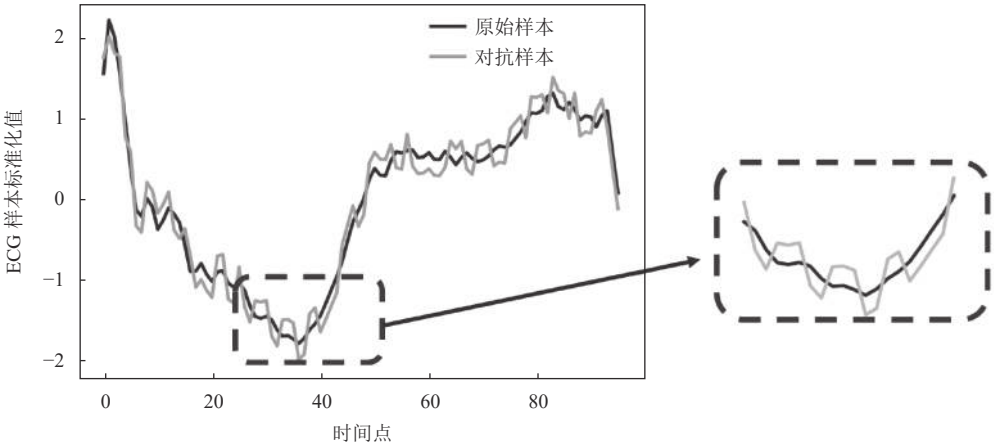


图 1 时间序列对抗攻击示例

为此本文提出一种局部扰动合并差分进化算法的半白盒攻击方法 (local perturbation merged differential evolution, AIMDE), 选用 FGSM 的改进版算法——基本迭代法 (basic iterative method, BIM)[11], 通过细化每一次攻击的梯度, 保证扰动幅度在一定范围内, 且为最优幅度. 之后利用滑动窗口结合时间序列预测模型的特性生成局部扰动序列, 从而大幅减少被感知的可能性. 此外, 我们采用差分进化算法 (differential evolution, DE)[12], 在没有任何目标区间和模型结构信息的情况下寻找最优的攻击点位, 目的是完成局部区间内的最佳扰动. 对于扰动幅度, 攻击

者可以通过增大或减小扰动因子的方式直接控制.

图 2 展示了 AIMDE (上方) 与 FGSM (下方) 的扰动样本对比, 灰色曲线为原始样本, 黑色曲线为对应方法生成的扰动样本. 从图 2 中看出, AIMDE 将扰动范围分散到局部区间中, 如图 2 上方的灰色虚线框所示. 对比基于全局攻击的对抗样本, AIMDE 与原始样本更为相近, 被感知度与全局扰动方法相比明显降低. 同时我们将对抗样本输入模型的预测攻击结果与真实预测结果的均方误差 (mean square error, *MSE*) 作为评判标准, *MSE* 差值越高说明预测偏离程度越大, 反之说明偏离程度越小.

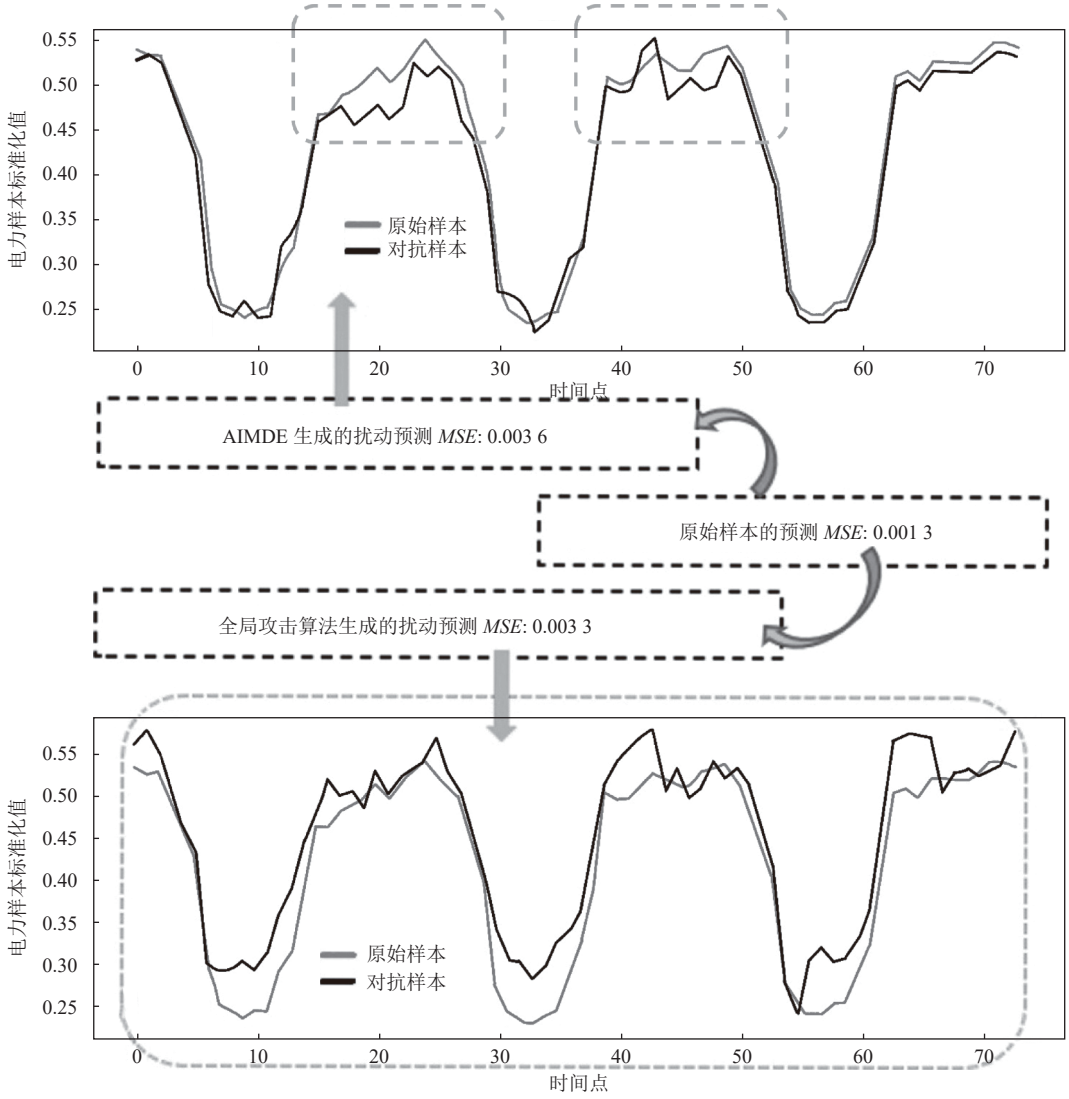


图 2　电力数据集的扰动样本对比示例图

从图 2 中可以看出, AIMDE 攻击预测结果的 *MSE* 差值最大 (0.0036>0.0033>0.0013). 因此, AIMDE 半白盒攻击方法具有以下优点:

1) 隐匿性: 攻击样本的扰动区间仅限于滑动窗口的局部间隔. 更少的扰动使它更不易被观察或检测出来.

2) 灵活性: AIMDE 可以通过调整扰动程度和不同的滑动窗口大小来建立不同的扰动区间, 同时由于差分进化算法的选择性, 相同区间下也能生成不同的扰动样本.

本文所提出方法生成的对抗样本在低感知度的情况下, 成功攻击在加州大学欧文分校 (University of California Irvine, UCI) 电力数据集[13]、标准普尔 500 指数 (Standard & Poor's, S&P 500) 股票价格数据集[14]、墨尔本每日最低气温数据集 (Temperature)[15]以及月平均太阳黑子数据集 (Sunspots)[16]上训练的 LSTM、TCN、Transformer[17]和 Informer[18]. 主要贡献如下.

1) 提出一种半白盒攻击方法, 在白盒全局迭代的梯度攻击下, 使用滑动窗口完成局部攻击, 并将滑动窗口内部比拟为黑盒从而寻求最优攻击点.

2) 所提出的方法是第 1 个通过向白盒攻击中添加差分进化算法进行区间最优选择的局部扰动方法.

3) 与全局扰动相比, AIMDE 生成的对抗样本在提升与原始实例相关系数的同时降低了可感知度, 并保持较好的攻击效果.

本文第 1 节简要介绍时间序列预测模型和相关的对抗样本生成方法. 第 2 节分详细介绍了我们提出的方法. 第 3 节对本文提出的方法在实际数据集上的表现进行评估, 并与使用蒙特卡罗估计方法进行攻击的结果进行对比. 第 4 节进行总结并对未来工作进行展望.

1 相关工作

本节主要介绍关于时间序列预测深度学习模型以及相关对抗攻击工作.

1.1 时间序列预测深度模型

深度模型在自然语言处理以及图像识别等领域取得了重大突破, 也可以用来进行时间序列预测, 其主要模型包括卷积神经网络 (convolutional neural network, CNN)、循环神经网络 (recurrent neural network, RNN)、TCN 和 Transformer 等. LSTM 模型因其能够学习长短期依赖信息关系, 非常适用于时间序列预测问题. Fischer 等人[19]使用 LSTM 在金融市场预测中取得了较好的效果. 同时, LSTM 也广泛应用于飞机发动机剩余使用寿命预测[20]、石油产量预测[21]以及太阳辐射预测[22]等. 此外, 具有代表性的 CNN 也同样适用于金融时间序列的预测[23]. Dong 等人[24]提出了一种基于 CNN 的词袋模型, 成功对某地区电力运转每小时的负载量进行预测. TCN 是一种新型的可以用来解决时间序列预测的算法, 其在金融时间序列预测[25]以及天气预报[26]等领域均取得了较好的效果. 除此之外, 近期也有一些比较先进的预测模型, 如 Zhou 等人[18]设计出一种基于 Transformer 的用于长序列时间序列预测的模型 Informer, 很大程度上提高了预测速度. Wu 等人[27]提出了一种用于时间序列预测的 Autoformer 模型, 其基于序列周期性的自相关机制, 使得模型在运行效率以及准确性上都得到提升.

1.2 时间序列预测对抗攻击

Szegedy 等人[5]首次提出了用于图像识别的对抗攻击, 从而引发了研究人员对不同领域对抗攻击的研究热潮. 目前图像领域的白盒攻击方法主要是基于模型梯度进行攻击, 如 FGSM 和 BIM 在图像识别任务中有着较好的攻击效果[10,11]. 图像领域的黑盒攻击方法主要基于替代模型和遗传算法. 例如, Su 等人[28]提出的像素攻击可以只通过改变图像中一个像素的信息来生成对抗样本. Papernot 等人[29]提出了一种利用替代模型的可转移性来逼近目标模型的黑盒方法. 此外, 基于生成对抗网络的 ANGRI 和 UPSET 模型[30]可以通过自监督模式训练网络来生成对抗样本.

然而, 面向时间序列的对抗样本生成方法较少. Oregi 等人[31]首次研究时间序列对抗攻击问题, 尽管他们只在模拟数据集上进行了实验, 但生成的样本可成功攻击 K 近邻分类器. Fawaz 等人[7]利用 FGSM 和 BIM 对残差网络 (residual network, ResNet)[31]进行攻击, 降低 ResNet 对时间序列分类的准确性. Rathore 等人[32]将白盒方法扩展到对时间序列有针对性和普适性的对抗攻击中. Karim 等人[33]提出梯度对抗变换网络 (gradient adversarial transform network, GATN) 来生成单变量和多变量时间序列[34]. 为了保持黑盒限制并获得梯度信息, 其方法需要通过知识蒸馏来训练替代模型. 针对时间序列预测的攻击工作较少, Dang-Nhu 等人[35]在概率自回归预测模型上提出了对时间序列预测的对抗攻击. Wu 等人[36]利用预测模型的梯度信息提出一种基于重要性度量的对抗时间序列样本白盒生成方法. 此外, Govindarajulu 等人[37]提出了一种定向、有幅度和以时间为目标的公式, 用于针对时间序列预测模型进行对抗攻击.

对比而言, 已有的白盒攻击方法 (如 FGSM 和 BIM) 基于整条序列进行扰动, 导致扰动样本容易被感知从而攻击失败. 黑盒攻击方法 (如 GATN) 需要频繁地调用原模型来训练替代模型, 这会提高被察觉的可能性. 而 AIMDE 不仅能够减少扰动范围, 提升对抗样本隐匿性, 还能通过 DE 算法避免多次调用模型内部信息去生成最终的扰动样本.

1.3 快速符号梯度算法和基本迭代算法

FGSM 通过获取训练中代价函数相对于模型输入的梯度, 并以此确定攻击方向, 生成对抗样本. 因为对抗扰动是通过单步计算生成的, 这种攻击也称为单步法. 生成对抗样本的公式如下:

$$\eta = \varepsilon \cdot sign(\nabla_x J_f(\boldsymbol{x}, y)) \tag{1}$$

$$\boldsymbol{x}^{\text{adv}} = \boldsymbol{x} + \eta \tag{2}$$

其中, J_f 是模型 f 的代价函数, ∇_x 表示模型相对于原始时间序列 $\boldsymbol{x}$ 的梯度, y 为正确的标签, ε 表示控制扰动幅度的超参数, $\boldsymbol{x}^{\text{adv}}$ 是时间序列的对抗样本.

BIM 是 FGSM 的改进算法, 由于 FGSM 的每步扰动都为固定大小, 意味着数据都会受到同样大小的扰动, 从而导致较大的扰动幅度. 因此 BIM 通过采用较小的步长对样本迭代添加扰动, 从而在产生更强对抗样本的同时保证每次扰动都在设定的范围内. 算法 1 展示了 BIM 的实现步骤, 它需要有 3 个超参数: 迭代次数 I, 每步的扰动幅度 α 和最大扰动幅度 ε. BIM 生成的对抗样本更接近原始样本, 被感知的概率更低.

算法 1. BIM 在时间序列预测上的攻击算法.

输入: 原始时间序列 $\boldsymbol{x}$ 和它的标签 y, 迭代次数 I, 每步扰动幅度 α, 最大扰动幅度 ε;
输出: 对抗样本 $\boldsymbol{x}^{\text{adv}}$.

$\boldsymbol{x}_0^{\text{adv}} \leftarrow \boldsymbol{x}$;
while $i = 0 \leqslant I$ **do**
  $\eta = \alpha \cdot sign(\nabla_x J_f(\boldsymbol{x}_i^{\text{adv}}, y))$; /*通过梯度方向生成扰动*/
  $\boldsymbol{x}_{i+1}^{\text{adv}} = \boldsymbol{x}_i^{\text{adv}} + \eta$;
  $\boldsymbol{x}_{i+1}^{\text{adv}} = \min\{\boldsymbol{x}+\varepsilon, \max\{\boldsymbol{x}-\varepsilon, \boldsymbol{x}_{i+1}^{\text{adv}}\}\}$; /*保证扰动在给定范围内*/
  i++;
end while

1.4 差分进化算法

差分进化算法是受自然选择过程启发的遗传算法的变体, 是一种基于种群的全局优化启发式方法[12]. 它可以解决多目标、多约束、大规模和不确定的优化问题[38]. 此外, 还可以在种群选择阶段保留个体的多样性, 从而比较容易找到更高质量的解决方案和全局最优解[39]. 与遗传算法类似, DE 有 3 个操作: 变异、交叉和选择. 图 3 中的黄色虚线框说明了 DE 算法的机制. 种群个体通过 DE 算法中的差分运算发生变异:

$$V_{i,g+1} = X_{r1,g} + F \cdot (X_{r2,g} - X_{r3,g}),\ i \neq r1 \neq r2 \neq r3 \tag{3}$$

其中, V、X 和 F 分别代表变异个体、父母和比例因子. i 代表种群中的第 i 个成员; $r1$、$r2$ 和 $r3$ 是随机个体; g 代表第 g 代. 之后, 变异成员 $V_{i,g+1}$ 和 $X_{i,g}$ 完成交叉的运算如下:

$$U_{j,i,g+1} = \begin{cases} V_{j,i,g+1}, & \text{if } rd_{j,i} \leqslant CR \text{ or } j = j_{rd} \\ X_{j,i,g}, & \text{otherwise} \end{cases} \tag{4}$$

其中, j 表示候选解的第 j 维变量. $rd_{j,i}$ 是一个介于 0 和 1 之间的随机值. CR 表示交叉的概率. j_{rd} 是一个随机的整数. U 是实验个体. 在选择阶段, 如果后代比相应的父母有更好的适应值, 则保留此后代. 否则, 保留父母. 选择公式为:

$$X_{i,g+1} = \begin{cases} U_{i,g+1}, & \text{if } q(U_{i,g+1}) \leqslant q(X_{i,g}) \\ X_{i,g}, & \text{otherwise} \end{cases} \tag{5}$$

其中, q 表示评估个体质量的适应度函数.

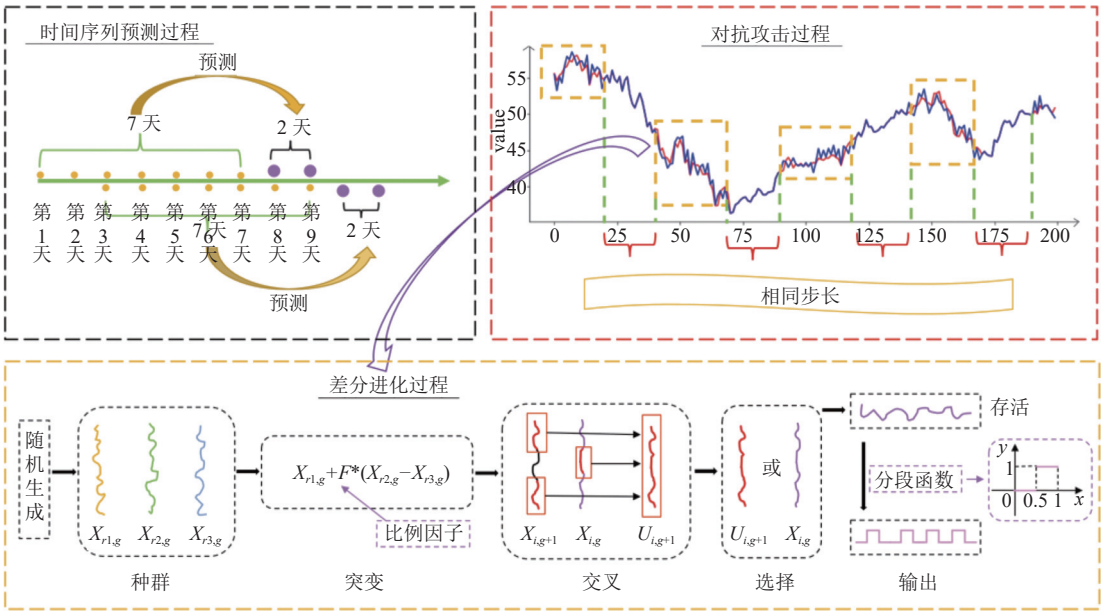


图 3 对抗攻击方法示意图

2 基于局部扰动与差分进化的半白盒攻击方法

在本节中, 首先提供问题的正式描述. 然后介绍如何实现半白盒攻击.

2.1 问题描述

假定攻击者可以在白盒设置中访问目标模型以及它的内部信息, 如参数, 网络架构和训练数据等. 因此可以使用目标训练集在训练过程中计算梯度并生成对抗样本. 相反, 在黑盒攻击中, 攻击者不能获取目标模型信息, 只能得到带有置信度或概率值的输出标签, 将随机生成的数据群体作为测试集, 根据模型反馈的标签, 不断优化群体中的个体置信度从而完成黑盒攻击. 本文提出的半白盒攻击方法将通过读取一次模型内部梯度信息获取全局扰动样本, 因滑动窗口内部信息无法获取, 所以在窗口内部使用黑盒攻击寻求最佳点位.

故生成对抗样本来攻击时间序列预测深度学习模型的攻击方法可以被形式化为约束优化问题. f 代表目标模型. 对于时间序列 $\boldsymbol{x}=(x_1,\ldots,x_n)$, 其中每个元素代表对应时间戳的数值. 设定原始时间序列 $\boldsymbol{x}$ 为 $\boldsymbol{x}_{\text{ori}}$, 则 f 预测值的 MSE 为 $f_{\text{ori}}(\boldsymbol{x})$. 类似地, 滑动窗口区间 l ($l \leqslant n$) 上的扰动可以表示为向量 $\epsilon(x)$.

在无目标攻击中, 攻击者为以下问题寻求优化解决方案:

$$\begin{cases} \max\limits_{\epsilon(x)^*} f_{\text{ori}}(\boldsymbol{x}+\epsilon(x)) \\ \text{s.t. } \|\epsilon(x)\|_0 \leqslant l,\ \|\epsilon(x)\|_\infty \leqslant \varepsilon \end{cases} \tag{6}$$

其中, $\|\epsilon(x)\|_0$ 和 $\|\epsilon(x)\|_\infty$ 分别表示 $\epsilon(x)$ 的 L_0 和 L_∞ 范数. 这意味着要添加扰动来最大化目标模型预测值的 MSE, 相当于最小化原模型的预测准确率.

对于有目标的攻击, 解决方案是最大程度接近给定目标 $\boldsymbol{x}_{\text{adv}}$ 的预测 MSE 值. 优化问题是:

$$\begin{cases} \min\limits_{\epsilon(x)^*} f_{\text{adv}} - f_{\text{ori}}(x+\epsilon(x)) \\ \text{s.t. } \|\epsilon(x)\|_0 \leqslant l,\ \|\epsilon(x)\|_\infty \leqslant \varepsilon \end{cases} \tag{7}$$

其中, f_{adv} 代表目标预测值的 MSE, 这就意味着要最小化被攻击模型预测值的 MSE 与目标 MSE 值的差距.

2.2 攻击方法

本文提出的攻击方法旨在满足上述约束条件的同时成功生成对抗样本. 为此, 根据时间序列预测的特点, 提

出 BIM 攻击结合滑动窗口和差分进化算法的半白盒攻击方法——AIMDE 来解决上述约束优化问题. 该方法主要分为两个阶段: 首先寻找合适的滑动窗口大小与步长, 之后根据差分进化算法生成最终对抗样本.

(1) 时间序列预测过程如图 3 黑色框中所示, 主要特点是根据设定的一段时间点 (1, 2,…, t) 通过模型训练去预测紧接着的一段固定时间 T (t+1, t+2,…, t+T) 的数据, 整个序列的预测应分为等长的预测时间段进行. 图 3 红色框内展现了基于局部攻击的思想, 利用滑动窗口来分割每个预测时间段, 根据时间序列预测的特点, 一个完整的扰动区间要与将被预测的时间点组成的区间长度成比例, 这样能够保证局部扰动均匀地分配到整个时间序列预测结果中, 所以滑动窗口的大小 W 以及步长 S 应满足:

$$W+S=\frac{T}{k},\ k=1,2,\ldots \tag{8}$$

(2) 算法 2 说明了在无目标攻击下对抗样本的生成过程. 首先通过 BIM 攻击算法获得全局攻击样本 (第 1 行), 直接获取全局样本的优势是可以省去在滑动窗口中重复进行 BIM 攻击的操作. 接下来根据公式 (8) 结合模型内部预测结构得到合适的滑动窗口大小 (第 2 行). 我们将滑动窗口内部空间比拟为一个黑盒环境, 使用差分进化算法进行攻击, 算法的具体流程如图 3 黄色框所示, 种群个体中的序列点随机从攻击样本和原始样本中抽取并组合为一条序列 (第 4–8 行). 接下来在每一代种群中, 孩子都是由上一代父母的变异和交叉后产生的, 并通过分段函数 h 转化为 0 和 1 的序列, 从而完成部分攻击样本的实现, 再进行个体的预测 MSE 值 f 计算, 并将结果作为个体的适应度. 对于更好的攻击样本而言, 预测结果 MSE 值越高的成员, 适应度就越高, 最终获胜者可以存活到下一代 (第 9–19 行). 其中最优个体同样会通过分段函数 h 进行 0 和 1 序列的转化并结合 BIM 作为算法的输出 (第 20–28 行), 当 y 取 0 表明当前序列点保留为原始样本数据, 当 y 取 1, 表明当前序列点采用 BIM 全局攻击样本中对应序列点的数据, 所以分段函数 h 应当满足:

$$y=\begin{cases}0, & 0\leqslant x<0.5\\ 1, & 0.5\leqslant x<1\end{cases} \tag{9}$$

分段函数存在的意义是对样本中序列点是否被攻击进行判断, 从而达到减少攻击范围的目的. 其思想来源于逻辑斯蒂回归函数, 将是否攻击问题转化为典型的二分类情况进行判断, 其中 0.5 作为判定指标出现在分段函数中, 当 $x\geqslant 0.5$ 时, y 取值为 1, 表明当前点位将被视作攻击点位, 当 $x<0.5$ 时, y 取值为 0, 当前时间序列点位将保留原始数值, 不会受到攻击. 因此, 分段函数的作用就是将差分进化得到的后代转化为是否对序列点位进行攻击的 0-1 序列, 之后结合原始样本从而产生新样本进行适应度计算.

尽管我们将攻击形式化为优化问题, 但不必在实践中找到最佳解决方案. 算法中给出的是无目标攻击, 需要运行完攻击者所设定的差分进化算法的进化代数, 之后存活的最优个体直接成为此次攻击样本, 当最优个体的预测准确率达到收敛时停止攻击 (第 21–23 行), 之后继续攻击其他预测区间, 直到获得完整的对抗攻击列表 (第 28 行). 如果是有目标攻击, 当最优个体的预测准确率大于或等于我们所给的目标 MSE 时, 则判断为攻击成功.

算法 2. 在无目标攻击下生成对抗样本.

输入: 时间序列数据集 $\boldsymbol{D}_{\text{ori}}$ 和被攻击的深度学习模型 M, ε 扰动因子的大小;
输出: 对抗样本.

1. *glb_adv_list* ← *GetGlobalAttack*($\boldsymbol{D}_{\text{ori}}$, ε , M); /*通过 BIM 梯度攻击得到全局攻击样本*/
2. W ← *FindingWindowSetting*(M); /*通过对模型的分析得到滑动窗口的大小*/
3. *adv_list* ← {};
4. **for** each $\boldsymbol{x}_{\text{ori}}$ in (*glb_adv_list* or $\boldsymbol{D}_{\text{ori}}$) **do** /*随机生成初始化种群个体*/
5. **for** i = 1, 2,…, n in *population* **do**
6. ϵ_i^1 = *init*(W, ε) ; /*初始化*/
7. **end for**
8. **for** g = 2,…, G *generation* **do**

for i = 1, 2,…, n in *population* **do**
  $v_i^g \leftarrow Mutation(\epsilon_i^{g-1})$; /*变异*/
  $u_i^g \leftarrow Crossover(v_i^g, \epsilon_i^{g-1})$; /*交叉*/
  if $f_{\text{ori}}(\boldsymbol{x}_{\text{ori}}+h(u_i^g)) < f_{\text{ori}}(\boldsymbol{x}_{\text{ori}}+h(\epsilon_i^{g-1}))$ **then**
    /*通过分段函数转化为时间序列进行适应度计算*/
    $\epsilon_i^g \leftarrow u_i^g$; /*选择更好的个体*/
  else
    $\epsilon_i^g \leftarrow \epsilon_i^{g-1}$;
  end if
  $p_i^g \leftarrow f_{\text{ori}}(\boldsymbol{x}_{\text{ori}}+h(\epsilon_i^g))$;
end for
$\varepsilon^* \leftarrow \epsilon^g_{\text{argmin}_j p_j^g}$; /*保留最佳个体*/
if argmax$f_c(\boldsymbol{x}_{\text{ori}}+h(\varepsilon^*)) \rightarrow$ *convergence* **then**
  $\boldsymbol{x}_{\text{adv}} \leftarrow \boldsymbol{x}_{\text{ori}} + h(\varepsilon^*)$; /*目标模型预测值的 *MSE* 达到收敛, 则为攻击成功*/
  adv_list.append($\boldsymbol{x}_{\text{adv}}$);
  break;
end if
end for
end for
return *adv_list*

3 实验与评价

实验数据集使用 UCI 数据库中的电力数据集 Electricity[13]、金融行业 S&P 500 股票价格数据集[14]以及来源自 Kaggle 数据库中的墨尔本每日最低气温数据集 Temperature[15]和月平均太阳黑子数据集 Sunspots[16], 详细内容列举在表 1 中, 4 者都来自时间序列预测领域中安全保障需求度较高的行业. 气温数据预测的研究使得航空航天、农业等产业能够掌握未来天气情况以便做好相应的应对措施. 太阳黑子由太阳磁场变动而产生, 与一些极端天气的出现密切相关, 对其预测能够在一定程度上保障人类活动以及农业发展. 对比算法方面, 我们将 AIMDE 与 FGSM、BIM 以及局部 BIM 攻击进行比对. 评估标准包括均方误差 *MSE*、平均绝对误差 (mean absolute error, *MAE*) 和相关系数.

表 1 数据集描述

数据集	实例数	属性数	数据长度	是否缺失值
Electricity	370	1	32 304	无
S&P 500	1	6	2 522	无
Temperature	1	2	3 650	无
Sunspots	1	2	2 820	无

1) 均方误差, *MSE* 衡量扰动后预测的结果与真实值的差异, 如下所示:

$$MSE = \frac{1}{T}\sum_{i=1}^{T}(x_{i,\text{ori}} - x_{i,\text{adv}})^2 \tag{10}$$

其中, $x_{i,\text{ori}}$ ($x_{i,\text{adv}}$) 表示原始 (对抗) 时间序列的第 i 个元素. *MSE* 越高, 说明结果偏离程度越大, 攻击效果越好. 另外, 对抗样本与真实样本间的 *MSE* 越小, 可感知性越低.

2) 平均绝对误差, *MAE* 衡量的是扰动后预测结果与样本真实值之间距离的平均值, 如下所示:

$$MAE = \frac{1}{T}\sum_{i=1}^{T}|x_{i,\mathrm{ori}} - x_{i,\mathrm{adv}}| \tag{11}$$

MAE 越高, 说明结果偏离程度越大, 攻击效果越好. 另外, 对抗样本与真实样本间的 *MAE* 越小, 可感知性越低. 使用两个评价指标 (*MAE* 和 *MSE*) 的目的是充分验证模型的优越性.

3) 相关系数[40], 评价扰动样本与真实样本之间的相关性. 如下所示:

$$r(\boldsymbol{x}_{\mathrm{ori}}, \boldsymbol{x}_{\mathrm{adv}}) = \frac{1}{T}\sum_{i=1}^{T}\frac{Cov(x_{i,\mathrm{ori}}, x_{i,\mathrm{adv}})}{\sqrt{Var(x_{i,\mathrm{ori}})Var(x_{i,\mathrm{adv}})}} \tag{12}$$

其中, *Cov* 表示 $x_{i,\mathrm{ori}}$ 和 $x_{i,\mathrm{adv}}$ 的协方差, *Var* 表示元素的方差. 扰动样本与原始样本的相关系数越大, 说明扰动样本与原始样本越贴近, 攻击的隐匿性越好, 反之攻击隐匿性差.

3.1 实验环境与设置

实验选择 LSTM[2]、TCN[3]、Transformer[17]以及 Informer[18]作为目标攻击模型. TCN[3]在 CNN 基础上的增加了结构创新, 其主要分为两个部分: 首先使用 CNN 对时空信息进行编码来计算低维嵌入表示, 之后将嵌入向量输入到 RNN 中. 相比 CNN, TCN 的设计是针对时间序列数据的, 所以选择更先进的 TCN 模型进行实验. Transformer 和 Informer 是近年来提出的先进时间序列预测模型. 实验使用 Adam 优化器[41]具有的默认超参数训练该模型, 以最小化交叉熵损失. 局部扰动的范围取决于滑动窗口的大小, 间隔是由步长决定. 为突出局部攻击的效果, 简单起见, 将滑动窗口与步长的比例 m 设置为{1, 2, 3}. 根据时间序列预测的特点, 滑动窗口大小与步长之和应与时间序列预测区间成正比. 对于 BIM 攻击设定迭代次数为 50 次, 每次迭代占总扰动的 2%. 此外, 扰动幅度 ε 的选取尤为重要, 其对攻击效果有着重要影响, 实验将 ε 的范围设置为{0.02, 0.04, 0.06, 0.08, 0.10}. 对于 DE 模型, 将公式 (3) 中的 F 和公式 (4) 中的 CR 交叉概率设置为默认值. 种群初始化使用默认的拉丁超立方体采样, 种群的大小设置为 600, 最大迭代次数为 60. 实验中, 对于有目标攻击, 在到达所设置的攻击目标 *MSE* 范围内后, 算法会提前结束迭代. 表 2 中总结出了所有需要考虑的参数.

表 2 参数设置

变量	值	描述
k	{2, 4}	时间序列预测区间与滑动窗口大小和步长之和之比
m	{1, 2, 3}	滑动窗口大小和步长之比
ε	{0.02, 0.04, 0.06, 0.08, 0.10}	扰动幅度
G	60	DE的最大迭代次数
P	600	DE的种群大小
F	0.5	DE的比例因子
CR	0.1	DE的交叉概率

3.2 参数分析

图 4 展示了在电力数据集下, 不同 k 和 m 对 LSTM 模型上攻击样本的生成和模型攻击效果的影响. 就对抗攻击的目标而言, 需要在预测结果 *MSE* 尽可能扩大的同时, 使得对抗样本与原始样本之间的差距尽量小, 也就是样本间的 *MSE* 尽可能降低. 所以预测结果 *MSE* 与样本间 *MSE* 之间的比值应当尽可能的大. 从图 4 中可以明显看出, k 的 3 个不同取值下, 黑色直方图 (即 m=1 时) 比值最高, 因此对于 m 的取值应当设定为 1. 在 m 同为 1 的情况下, 要突出局部的效果, 所以 k 取值为 2.

图 5 展示了不同扰动幅度 ε 的取值在电力数据集上对样本的扰动以及对预测结果的影响. 在保证样本之间的 *MSE* 尽量小的前提下, 使预测结果的 *MSE* 更大, 故应选取预测结果 *MSE* 与样本之间 *MSE* 差值最大的 ε 取值. 当扰动大小取为 0.06 时, 差距最大, 因此我们设定 ε 的取值为 0.06.

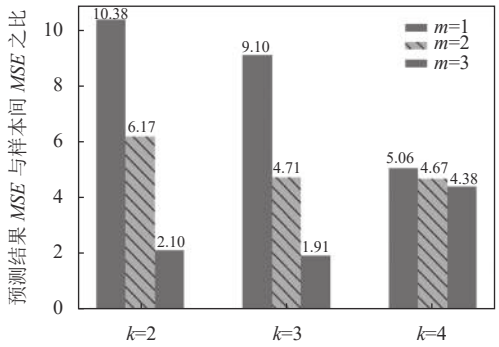


图 4 不同 k 和 m 下的预测结果 *MSE* 与样本间 *MSE* 比值统计图

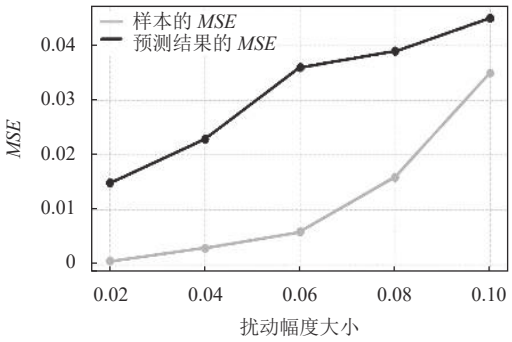


图 5 不同扰动幅度 ε 的影响示例图

3.3 实验对比

由于 FGSM 和 BIM 以及蒙特卡罗[35]等白盒攻击方法无法进行有目标的攻击, 所以本文主要进行无目标攻击下的对比. 我们在相同实验环境下, 使用相同的方法训练模型, 在相同参数的设定下对全局扰动方法, 局部扰动方法和 AIMDE 进行对比. 表 3 和表 4 中分别展示出了几种算法在电力数据集、股票数据集、气温数据集以及太阳黑子数据集上所得预测结果 *MSE* 和 *MAE* 的对比情况. 同时将未受攻击前的模型预测结果 *MSE/MAE* 放在最后作为实验的对比基准.

表 3 不同攻击方法下预测结果的 *MSE* 值

方法	数据集	FGSM	BIM	局部BIM	AIMDE	原始预测
LSTM	S&P 500	0.001 0	0.001 0	0.001 4	**0.001 5**	0.000 3
	Electricity	0.003 6	0.007 3	0.007 7	**0.007 8**	0.001 5
	Temperature	0.005 4	0.005 4	0.006 4	**0.008 4**	0.008 3
	Sunspots	0.005 3	0.002 6	0.005 1	**0.012 8**	0.011 9
TCN	S&P 500	0.000 7	**0.001 1**	0.000 2	0.000 4	0.000 2
	Electricity	0.001 9	0.002 0	0.002 2	**0.002 7**	0.001 6
	Temperature	0.009 6	0.006 7	0.006 9	**0.010 5**	0.007 1
	Sunspots	0.000 3	0.000 4	0.000 7	**0.003 9**	0.003 6
Transformer	S&P 500	**0.001 4**	**0.001 4**	0.000 9	0.000 9	0.000 1
	Electricity	0.001 7	0.001 7	0.001 7	**0.002 8**	0.001 6
	Temperature	0.014 3	0.014 3	0.014 5	**0.014 9**	0.010 6
	Sunspots	0.010 2	0.007 6	0.007 3	**0.010 6**	0.008 7
Informer	S&P 500	0.000 2	**0.000 9**	0.000 7	0.000 7	0.000 2
	Electricity	0.002 2	0.002 4	0.002 3	**0.002 8**	0.002 0
	Temperature	0.003 3	0.003 0	0.003 2	**0.006 4**	0.004 2
	Sunspots	0.005 3	0.005 4	0.005 2	0.005 4	**0.005 6**

如表 3 和表 4 所示, 粗体数据代表攻击方法在数据集上预测结果 (*MSE/MAE*) 的最大值. 在 Electricity 数据集上, AIMDE 使 4 种模型预测的 *MSE* 和 *MAE* 都高于其他的对比方法. 与原始预测对比, AIMDE 方法下的预测结果 *MSE* 有明显提升, 特别是在 LSTM 模型上, 使得 *MSE* 相比原始预测提升了 4 倍, *MAE* 提升约 1.2 倍. 在 S&P 500 股票数据集上, 除了 LSTM 模型, 全局攻击算法 (如 BIM 和 FGSM) 达到的攻击效果较好. 这是由于股票走势本身随机性大, 训练好的模型易受到微小扰动的干扰, 因此全局扰动在这种情况下表现更好. 另外, 对于 Transformer 和 Informer 模型, S&P 500 和 Electricity 数据集的 AIMDE 与全局 BIM 扰动预测结果的 *MSE* 相差不大. 但在样本的隐匿性方面, AIMDE 明显优于其他攻击方法. 在 Temperature 数据集上, AIMDE 对于 4 种模型预测结果的 *MSE/MAE* 达到最大, 表明其攻击效果最佳. 对于 Sunspots 数据集, 除了 AIMDE, 其余 3 种攻击方法在 LSTM、

TCN 和 Transformer 模型上并没起到攻击效果, 反而预测结果的 *MSE*/*MAE* 数值相比原始预测降低, 达到了数据增强效果, 但是在 Informer 模型上, 4 种攻击方法均未攻击成功. 如表 5 所示, AIMDE 生成的对抗样本与原始样本的相关系数在所有攻击模型上表现都是最优的, 这说明 AIMDE 在隐匿性上取得了显著的改进. 样本差别的具体示例图将在第 3.4 节中展示.

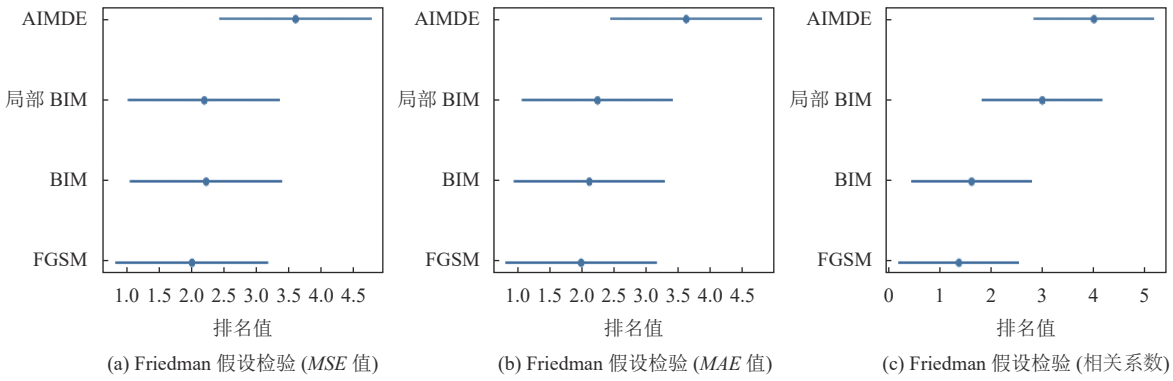
表 4　不同攻击方法下预测结果的 *MAE* 值

方法	数据集	FGSM	BIM	局部BIM	AIMDE	原始预测
LSTM	S&P 500	0.0306	0.0306	0.0375	**0.0379**	0.0173
	Electricity	0.0462	0.0658	0.0677	**0.0683**	0.0296
	Temperature	0.0562	0.0562	0.0640	**0.0730**	0.0727
	Sunspots	0.0705	0.0477	0.0693	**0.1116**	0.1078
TCN	S&P 500	0.0235	**0.0304**	0.0132	0.0172	0.0119
	Electricity	0.0294	0.0300	0.0375	**0.0376**	0.0269
	Temperature	0.0839	0.0648	0.0642	**0.0864**	0.0657
	Sunspots	0.0131	0.0151	0.0219	**0.0605**	0.0573
Transformer	S&P 500	**0.0342**	**0.0342**	0.0266	0.0270	0.0103
	Electricity	0.0352	0.0353	0.0332	**0.0459**	0.0280
	Temperature	0.1096	0.1096	0.1103	**0.1120**	0.0888
	Sunspots	0.0997	0.0858	0.0844	**0.1021**	0.0923
Informer	S&P 500	0.0123	0.0209	0.0184	**0.0167**	0.0123
	Electricity	0.0361	0.0373	0.0379	**0.0412**	0.0437
	Temperature	0.0507	0.0464	0.0503	**0.0726**	0.0590
	Sunspots	0.0660	0.0633	0.0666	0.0668	**0.0700**

表 5　不同方法的对抗样本与原始样本的相关系数统计结果

方法	数据集	FGSM	BIM	局部BIM	AIMDE
LSTM	S&P 500	0.8917	0.9474	0.9750	**0.9792**
	Electricity	0.9084	0.9128	0.9427	**0.9674**
	Temperature	0.8746	0.8694	0.9392	**0.9697**
	Sunspots	0.9346	0.9391	0.9637	**0.9770**
TCN	S&P 500	0.8413	0.8116	0.8745	**0.9412**
	Electricity	0.9091	0.9085	0.9385	**0.9649**
	Temperature	0.8456	0.8619	0.9302	**0.9748**
	Sunspots	0.9554	0.9516	0.9561	**0.9708**
Transformer	S&P 500	0.8737	0.8737	0.8826	**0.9101**
	Electricity	0.8638	0.8644	0.9372	**0.9646**
	Temperature	0.8561	0.8561	0.9373	**0.9606**
	Sunspots	0.9135	0.9324	0.9613	**0.9711**
Informer	S&P 500	0.7793	0.7890	0.8136	**0.9273**
	Electricity	0.9583	0.9502	0.9616	**0.9790**
	Temperature	0.8984	0.9180	0.9392	**0.9679**
	Sunspots	0.9017	0.9195	0.9550	**0.9797**

在表 3–表 5 的数据支持下, 我们对这 4 种攻击算法的差异性进行 Friedman 假设检验, 如图 6 所示, 方法之间若没有重叠区域, 则证明两个算法存在显著性差异. 其中图 6(a) 和图 6(b) 分别对应的是攻击算法在 *MSE* 和 *MAE* 下的差异性展示, AIMDE 与其余 3 种方法重叠区域较小, 说明 AIMDE 与 3 种方法达到的攻击效果有较大差异. 图 6(c) 对应的是不同攻击算法在对抗样本与原始样本相关系数下的差异性展示, AIMDE 与局部 BIM 算法图线部分重叠, 说明二者差异性不明显. 此外 AIMDE 与 BIM 和 FGSM 几乎没有重叠, 表明 AIMDE 在隐匿性上显著优于这两种攻击方法.



(a) Friedman 假设检验 (*MSE* 值) (b) Friedman 假设检验 (*MAE* 值) (c) Friedman 假设检验 (相关系数)

图 6 不同攻击方法的 Friedman 假设检验结果

此外, 我们将对比不同扰动幅度 ε 对 *MSE* 的影响. 如图 7 所示, 当 ε 逐步增大时, 3 种对比算法的预测 *MSE* 均呈上升趋势, 而 AIMDE 与局部 BIM 和全部 BIM 间的差异也逐渐扩大. 由于 AIMDE 基于局部扰动, 导致其生成样本与真实样本的 *MSE* 较小, 因此, 在低 ε 的情况下 (例如 0.02), AIMDE 的攻击效果并不会明显超过全局扰动. 对于具有灵活性的 DE 算法, 可以通过加大迭代次数以及种群数量去削弱扰动幅度过小的影响, 但是会增加计算的复杂度, 加重训练负担.

接下来, 探究不同 ε 取值对时间效率影响. 如图 8 所示, 随着扰动大小的改变, 3 种算法的时间效率没有显著的变化. 因此, AIMDE 攻击方法的优点之一是扰动幅度的变化不会对时间效率产生影响. 另外, 由于局部 BIM 是在全局 BIM 的基础上进行基于滑动窗口的扰动, AIMDE 在局部 BIM 的基础上通过 DE 算法寻找最优攻击点位, 三者运行效率的排序为 BIM > 局部 BIM > AIMDE.

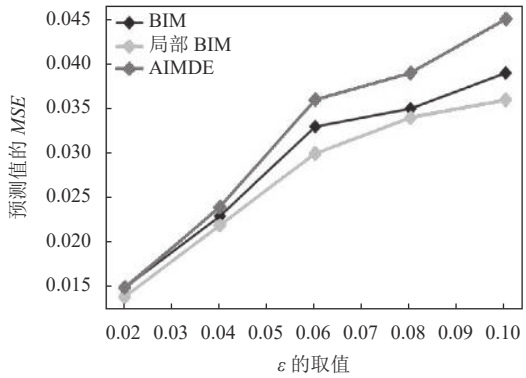


图 7 不同扰动幅度 ε 对 *MSE* 影响示例图

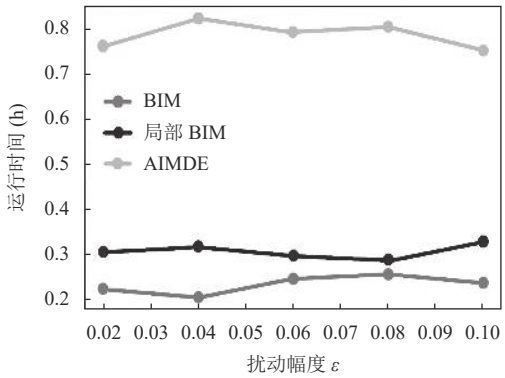


图 8 不同扰动幅度 ε 对运行时间的影响示例图

3.4 案例分析

此节通过结果图进行方法对比. 全局攻击、局部攻击和 AIMDE 的实验环境以及变量设定相同. 由于蒙特卡罗估计方法[35]与 AIMDE 在实验上无法完全设定相同环境与变量, 这里将复现结果进行对比展示.

图 9 展示了在最优扰动幅度 ε 取为 0.06 情况下所截取的样本部分区间结果的对比, 图中红色曲线为扰动样本, 蓝色曲线为原始样本, r 表示两者之间的相关系数. 实验分别对比了全局攻击 BIM, 局部攻击 BIM 和 AIMDE 所生成的对抗样本效果. 从人眼感官的角度可以看出, 无论在电力数据集还是股票数据集的全局扰动样本中, 都会存在一个较明显的波动, 导致容易被感知. 绿色虚线框内的对比表明局部扰动会降低扰动的范围, 但损失了扰动效果. 对于 AIMDE 而言, 局部扰动不仅降低了扰动范围, 而且滑动窗口内部实现的差分进化算法也保证了攻击效果. 此外 AIMDE 在电力数据集上的扰动可感知性比股票数据集更低, 这是由于模型在股票数据集上进行预测时, 为保证预测

精度而降低了扰动范围, 在损失一定攻击范围的前提下, 只能在有限的攻击范围内尽可能扩大攻击点, 以保证攻击效果.

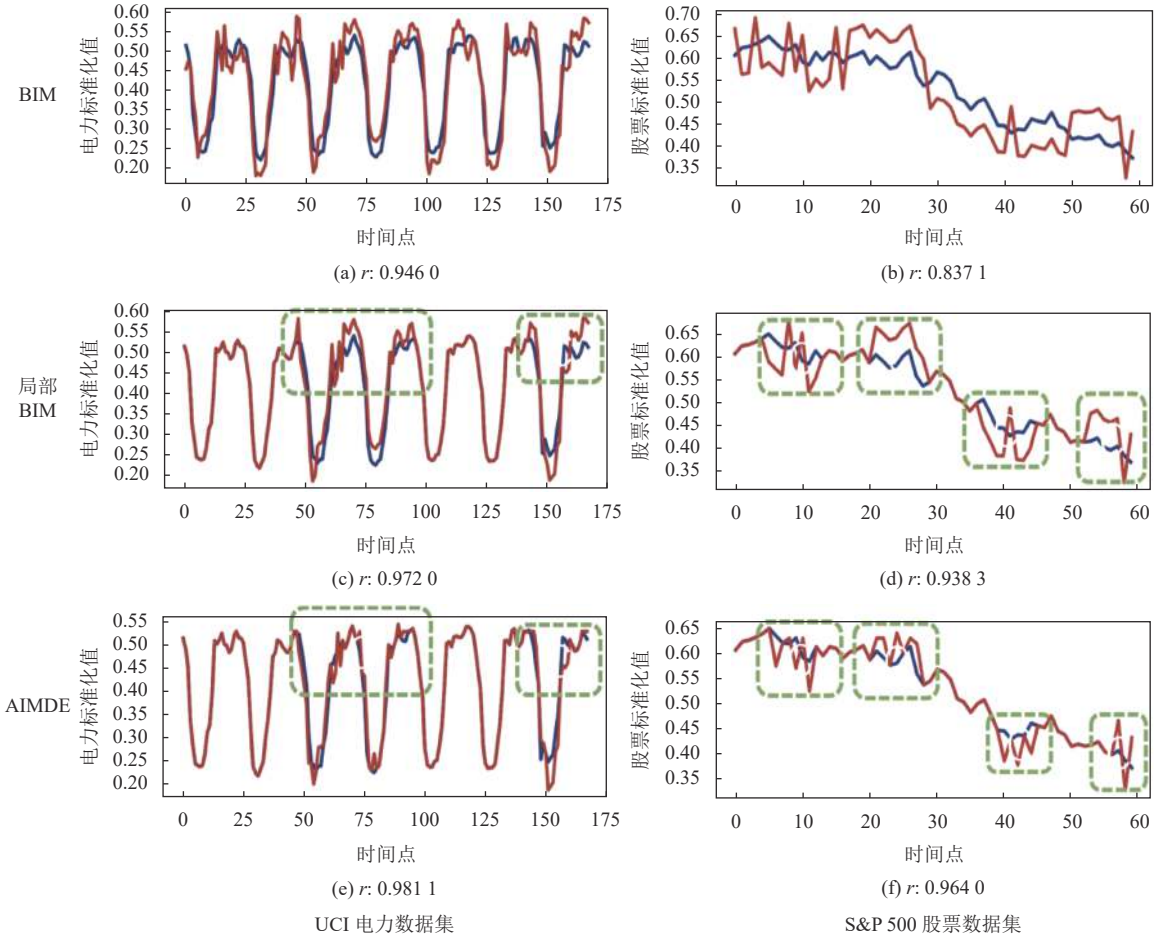


图 9 电力和股票数据对抗样本示例图 (ε=0.06)

图 10 展示了在最优扰动幅度 ε 取为 0.06 情况下不同对抗样本的预测结果. 图中红色线条代表标签的真实值, 蓝色线条为原始样本输入到模型 LSTM 中得到的预测值, 绿色线条为全局扰动 BIM 样本输入到模型得到的预测值, 黄色线条为局部攻击 BIM 样本输入到模型的预测值, 紫色线条为 AIMDE 生成的扰动样本输入到模型中的预测值. 对于 UCI 电力数据集, 选取了不同扰动方向的 4 张攻击结果图, 从黑色虚线框中看出, 全局攻击和局部攻击的结果较为接近且扰动范围有限, 扰动方向比较单一, 而 AIMDE 通过滑动窗口内部的差分进化选择, 保证了攻击的多样化与灵活性.

图 10 中最后一列图展示了股票数据集的结果, 由于股票的走势具有无规律性, 为了保证模型预测的有效性, 模型只对接下来 5 天的走势进行预测. 从黑色框中可以观察到, AIMDE 攻击导致的预测偏移程度没有全局扰动明显, 这种情况在预期之内. 使用局部扰动的原因是为了提高隐匿性, 但是会损失一部分攻击范围, 同时模型需要提高精确度来降低预测范围, 故 AIMDE 会在比局部扰动效果更好的前提下, 尽可能接近全局扰动结果.

图 11 展示了 AIMDE (下方) 与蒙特卡罗估计方法 (上方) 的结果对比. 图中绿色虚线左侧部分代表原始样本 (红色) 和对抗样本 (蓝色), 右侧部分的曲线代表原始样本输入模型的预测结果曲线 (红色) 和对抗样本输入模型的预测结果曲线 (蓝色). 从图中看出, AIMDE 在对抗样本的隐匿性上有较大提升, 同时攻击效果也保持稳定. 观察绿

色虚线右侧, AIMDE 会导致预测的峰值偏移, 且攻击效果比蒙特卡罗估计方法更好.

3.5 有目标攻击

对于时间序列预测的有目标攻击, 设定不同增幅的局部攻击 *MSE* 为自变量. 实验表明, AIMDE 可以在一定范围内进行有目标攻击, 统计结果如图 12 所示, 扰动成功率在增幅为 10% 以内较高, 随着增幅提高, 成功率会明显下降. 如果在高增幅的限制下提高成功率, 加大种群数量和迭代次数是可行方案, 也可以通过扩大扰动区间以及增加扰动幅度, 但这样会导致更复杂的计算, 也会增大被感知的可能性.

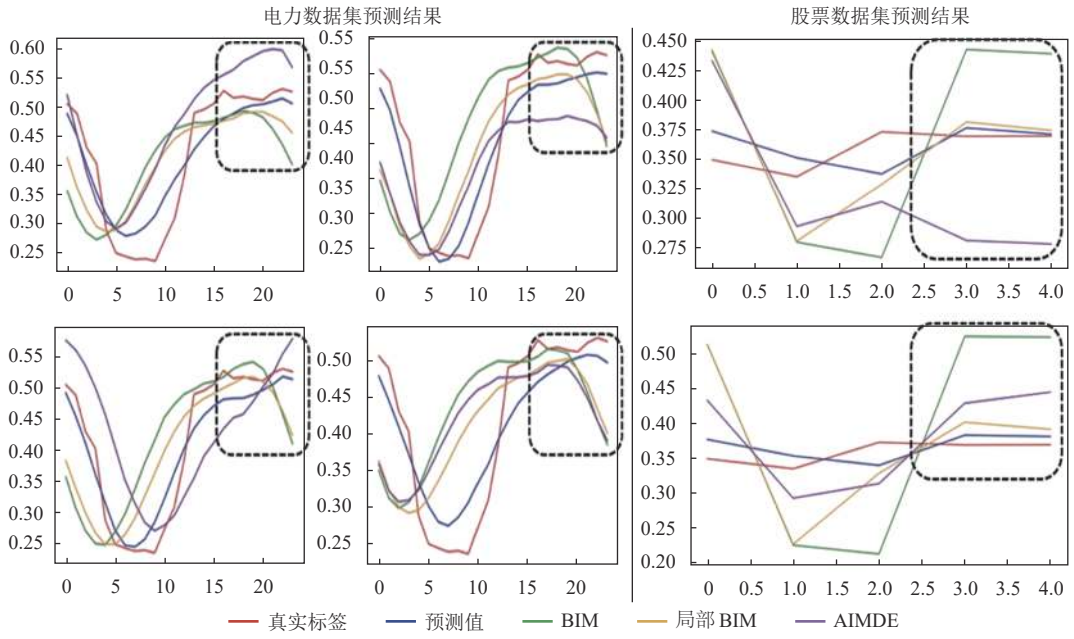


图 10 不同攻击方法的预测结果对比示例图

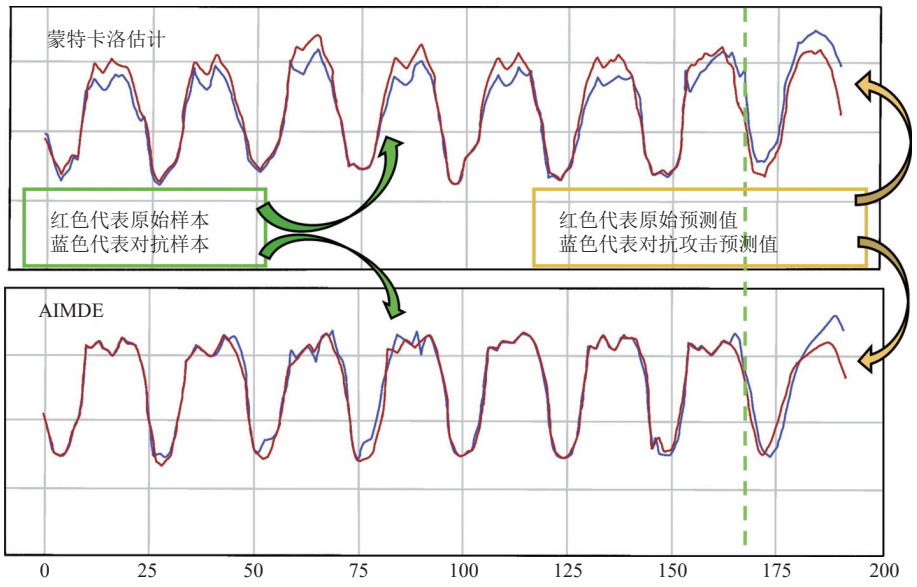


图 11 AIMDE 和蒙特卡罗实验结果对比示意图

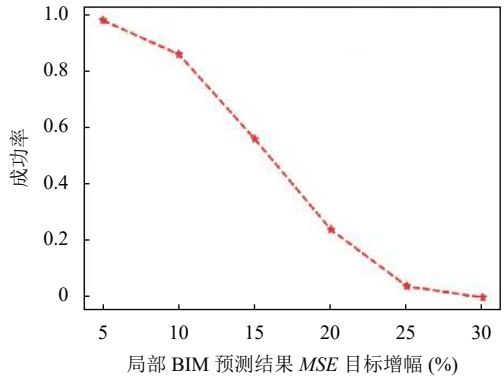


图 12　有目标攻击的成功率统计图

3.6　防御方法

研究对抗攻击的反制——防御策略是有必要的, 我们采用对抗训练[42]作为防御策略. 首先攻击 LSTM 并生成对抗样本, 记录扰动的 *MSE*, 之后将对抗样本加入到训练集中, 重新进行训练. 如表 6 所示, 对抗样本的 *MSE* 明显降低, 说明防御策略有效. 在此防御模型上继续进行全局攻击、局部攻击以及 AIMDE 攻击, 结果显示全局攻击和局部攻击效果不佳, 因为这两种方法不具备灵活性, 攻击比较固定. 而 AIMDE 仍然取得较好的攻击效果, 说明对抗防御训练并不能对 AIMDE 攻击算法提高模型的鲁棒性, 表明 AIMDE 生成的对抗样本具有多样性与攻击稳定性.

AIMDE 加入了差分进化算法, 其针对每一个滑动窗口内部的攻击点都是随机选择的, 因此生成的对抗样本具有随机性, 从而导致对抗训练策略失效. 一个针对性的策略是多重对抗防御训练[43], 即多次重复生成对抗样本, 利用这些对抗样本进行重复对抗训练, 能够有效地通过大量对抗样本弥补无法确定随机选择时间点的缺陷. 此外, 由于是针对有规律的数据集, 一些转折处的时间点权重不会发生明显变化, 导致随机选择的范围不会很广泛, 多重训练也不会造成大量的资源负担, 接下来通过实验进行测试.

如图 13 所示, *r* 代表对抗样本数量, 使用有规律电力数据集在扰动幅度大小为 0.6 情况下生成. 从图中可以看出, 当对抗样本数量达到 30 以上时, 针对 AIMDE 攻击的防御效果已经明显有效, 而且 30 次算法的迭代生成并不会造成大量的计算负担. 因此针对此数据集, 本节提出的多重对抗训练防御方法能有效针对 AIMDE 攻击进行防御.

表 6　不同攻击方法的对抗训练预测结果

方法	首次攻击预测结果*MSE*	防御之后预测*MSE*	再次攻击后的预测*MSE*
BIM攻击	0.003 3	0.001 5	0.001 8
局部BIM攻击	0.003 0	0.001 1	0.001 6
AIMDE攻击	**0.003 6**	0.001 2	**0.003 5**

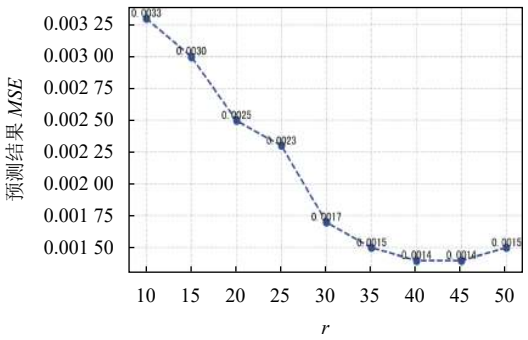


图 13　不同对抗样本数量下结果对比示意图

下面进行 AIMDE 攻击下的多重对抗训练防御策略是否具有普适性的实验, 主要针对 FGSM, BIM 和局部 BIM 进行测试. 测试过程中沿用之前实验的参数设置, 实验结果如表 7 所示, 通过多重对抗性训练后, LSTM 模型

的鲁棒性对于 3 种不同的攻击算法都有不同程度的提升, 但对 FGSM 算法的防御能力较差, 因为 FGSM 算法的生成过程与 BIM 不同, 攻击时间点的敏感程度也不如 BIM 算法. 多重训练使得模型进行了丰富的攻击点位学习, 所以针对 BIM 和局部 BIM 攻击有更好的防御性.

表 7　多重对抗训练的普适性

攻击方法	对抗训练前预测*MSE*	对抗训练后预测*MSE*
FGSM	0.002 8	0.002 4
BIM	0.003 3	0.001 6
局部BIM	0.003 1	0.001 4

4 结论和未来的工作

本文提出了一种半白盒时间序列预测攻击方法. 与之前需要全局扰动的研究不同, 本文基于滑动窗口和差分进化算法来实现局部扰动, 攻击时间序列预测深度学习模型. 实验结果表明, 所提出的算法可以有效攻击 UCI 电力数据集和 S&P 500 股票数据集, 并能够灵活生成扰动样本. 与目前主流的白盒攻击方法相比, AIMDE 不仅明显提升了对抗样本的隐匿性, 而且保证了最终扰动的效果. 未来的研究工作包括两个方面, 一是进一步研究如何提高攻击方法的效率; 二是针对所提出的攻击方法, 研究更有效的主动防御策略.

References:

[1] Yuan JD, Wang ZH. Review of time series representation and classification techniques. Computer Science, 2015, 42(3): 1–7 (in Chinese with English abstract). [doi: 10.11896/j.issn.1002-137X.2015.3.001]

[2] Hochreiter S, Schmidhuber J. LSTM can solve hard long time lag problems. In: Proc. of the 9th Int'l Conf. on Neural Information Processing Systems. Denver: MIT Press, 1997. 473–479.

[3] Lea C, Vidal R, Reiter A, Hager GD. Temporal convolutional networks: A unified approach to action segmentation. In: Proc. of the 2016 European Conf. on Computer Vision. Amsterdam: Springer, 2016. 47–54. [doi: 10.1007/978-3-319-49409-8_7]

[4] Lim B, Zohren S. Time-series forecasting with deep learning: A survey. Philosophical Trans. of the Royal Society A: Mathematical, Physical and Engineering Sciences, 2021, 379(2194): 20200209. [doi: 10.1098/rsta.2020.0209]

[5] Szegedy C, Zaremba W, Sutskever I, Bruna J, Erhan D, Goodfellow IJ, Fergus R. Intriguing properties of neural networks. In: Proc. of the 2nd Int'l Conf. on Learning Representations. Banff: ICLR, 2014.

[6] Wei CX, Wang ZH, Yuan JD, Lin QH. Time series pattern discovery and classification with variable scales in time-frequency domains. Ruan Jian Xue Bao/Journal of Software, 2022, 33(12): 4411–4428 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/6346.htm [doi: 10.13328/j.cnki.jos.006346]

[7] Fawaz HI, Forestier G, Weber J, Idoumghar L, Muller PA. Adversarial attacks on deep neural networks for time series classification. In: Proc. of the 2019 Int'l Joint Conf. on Neural Networks (IJCNN). Budapest: IEEE, 2019. 1–8. [doi: 10.1109/IJCNN.2019.8851936]

[8] Gasparin A, Lukovic S, Alippi C. Deep learning for time series forecasting: The electric load case. CAAI Trans. on Intelligence Technology, 2022, 7(1): 1–25. [doi: 10.1049/cit2.12060]

[9] Chen HX, Huang CY, Huang QY, Zhang Q, Wang W. ECGadv: Generating adversarial electrocardiogram to misguide arrhythmia classification system. In: Proc. of the 34th AAAI Conf. on Artificial Intelligence. 2020. 3446–3453. [doi: 10.1609/aaai.v34i04.5748]

[10] Goodfellow IJ, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. In: Proc. of the 3rd Int'l Conf. on Learning Representations. San Diego: ICLR, 2015.

[11] Kurakin A, Goodfellow IJ, Bengio S. Adversarial examples in the physical world. In: Proc. of the 5th Int'l Conf. on Learning Representations. Toulon: OpenReview.net, 2017.

[12] Storn R, Price K. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization, 1997, 11(4): 341–359. [doi: 10.1023/A:1008202821328]

[13] Blake C. UCI repository of machine learning databases. 1998. http://www.ics.uci.edu/mlearn/MLRepository.html

[14] Sitte R, Sitte J. Analysis of the predictive ability of time delay neural networks applied to the S&P 500 time series. IEEE Trans. on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 2000, 30(4): 568–572. [doi: 10.1109/5326.897083]

[15] Brabban P. Daily minimum temperatures in Melbourne. 2018. https://www.kaggle.com/datasets/paulbrabban/daily-minimum-temperatures-in-melbourne

[16] Paul S. Sunspots. 2020. https://www.kaggle.com/datasets/robervalt/sunspots

[17] Wu N, Green B, Ben X, O'Banion S. Deep transformer models for time series forecasting: The influenza prevalence case. arXiv: 2001.08317, 2020.

[18] Zhou HY, Zhang SH, Peng JQ, Zhang S, Li JX, Xiong H, Zhang WC. Informer: Beyond efficient transformer for long sequence time-series forecasting. In: Proc. of the 35th AAAI Conf. on Artificial Intelligence. AAAI Press, 2021. 11106–11115. [doi: 10.1609/aaai.v35i12.17325]

[19] Fischer T, Krauss C. Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research, 2018, 270(2): 654–669. [doi: 10.1016/j.ejor.2017.11.054]

[20] Yuan M, Wu YT, Lin L. Fault diagnosis and remaining useful life estimation of aero engine using LSTM neural network. In: Proc. of the 2016 IEEE Int'l Conf. on Aircraft Utility Systems (AUS). Beijing: IEEE, 2016. 135–140. [doi: 10.1109/AUS.2016.7748035]

[21] Sagheer A, Kotb M. Time series forecasting of petroleum production using deep LSTM recurrent networks. Neurocomputing, 2019, 323: 203–213. [doi: 10.1016/j.neucom.2018.09.082]

[22] Sorkun MC, Incel OD, Paoli C. Time series forecasting on multivariate solar radiation data using deep learning (LSTM). Turkish Journal of Electrical Engineering and Computer Sciences, 2020, 28(1): 211–223. [doi: 10.3906/elk-1907-218]

[23] Arratia A, Sepúlveda E. Convolutional neural networks, image recognition and financial time series forecasting. In: Proc. of the 4th Workshop on Mining Data for Financial Applications. Würzburg: Springer, 2019. 60–69. [doi: 10.1007/978-3-030-37720-5_5]

[24] Dong XS, Qian LJ, Huang L. A CNN based bagging learning approach to short-term load forecasting in smart grid. In: Proc. of the 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation. San Francisco: IEEE, 2017. 1–6. [doi: 10.1109/UIC-ATC.2017.8397649]

[25] Borovykh A, Bohte S, Oosterlee CW. Dilated convolutional neural networks for time series forecasting. Journal of Computational Finance, 2019, 22(4): 73–101. [doi: 10.21314/JCF.2019.358]

[26] Yan JN, Mu L, Wang LZ, Ranjan R, Zomaya AY. Temporal convolutional networks for the advance prediction of ENSO. Scientific Reports, 2020, 10(1): 8055. [doi: 10.1038/s41598-020-65070-5]

[27] Wu HX, Xu JH, Wang JM, Long MS. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. arXiv:2106.13008, 2021.

[28] Su JW, Vargas DV, Sakurai K. One pixel attack for fooling deep neural networks. IEEE Trans. on Evolutionary Computation, 2019, 23(5): 828–841. [doi: 10.1109/TEVC.2019.2890858]

[29] Papernot N, McDaniel P, Goodfellow I. Transferability in machine learning: From phenomena to black-box attacks using adversarial samples. arXiv:1605.07277, 2016.

[30] Sarkar S, Bansal A, Mahbub U, Chellappa R. UPSET and ANGRI: Breaking high performance image classifiers. arXiv:1707.01159, 2017.

[31] Oregi I, Del Ser J, Perez A, Lozano JA. Adversarial sample crafting for time series classification with elastic similarity measures. In: Proc. of the 2018 Int'l Symp. on Intelligent and Distributed Computing. Cham: Springer, 2018. 26–39. [doi: 10.1007/978-3-319-99626-4_3]

[32] Rathore P, Basak A, Nistala SH, Runkana V. Untargeted, targeted and universal adversarial attacks and defenses on time series. In: Proc. of the 2020 Int'l Joint Conf. on Neural Networks (IJCNN). Glasgow: IEEE, 2020. 1–8. [doi: 10.1109/IJCNN48605.2020.9207272]

[33] Karim F, Majumdar S, Darabi H. Adversarial attacks on time series. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2021, 43(10): 3309–3320. [doi: 10.1109/TPAMI.2020.2986319]

[34] Harford S, Karim F, Darabi H. Adversarial attacks on multivariate time series. arXiv:2004.00410, 2020.

[35] Dang-Nhu R, Singh G, Bielik P, Vechev M. Adversarial attacks on probabilistic autoregressive forecasting models. In: Proc. of the 37th Int'l Conf. on Machine Learning. JMLR.org, 2020. 2356–2365.

[36] Wu T, Wang XC, Qiao SJ, Xian XP, Liu YB, Zhang L. Small perturbations are enough: Adversarial attacks on time series prediction. Information Sciences: An Int'l Journal, 2022, 587: 794–812. [doi: 10.1016/j.ins.2021.11.007]

[37] Govindarajulu Y, Amballa A, Kulkarni P, Parmar M. Targeted attacks on timeseries forecasting. arXiv:2301.11544, 2023.

[38] Das S, Suganthan PN. Differential evolution: A survey of the state-of-the-art. IEEE Trans. on Evolutionary Computation, 2010, 15(1): 4–31. [doi: 10.1109/TEVC.2010.2059031]

[39] Civicioglu P, Besdok E. A conceptual comparison of the cuckoo-search, particle swarm optimization, differential evolution and artificial

bee colony algorithms. Artificial Intelligence Review, 2013, 39(4): 315–346. [doi: 10.1007/s10462-011-9276-0]

[40] Schober P, Boer C, Schwarte LA. Correlation coefficients: Appropriate use and interpretation. Anesthesia & Analgesia, 2018, 126(5): 1763–1768. [doi: 10.1213/ANE.0000000000002864]

[41] Kingma DP, Ba LJ. Adam: A method for stochastic optimization. In: Proc. of the 2015 Int'l Conf. on Learning Representations (ICLR). Ithaca: ICLR, 2015.

[42] Madry A, Makelov A, Schmidt L, Tsipras D, Vladu A. Towards deep learning models resistant to adversarial attacks. In: Proc. of the 6th Int'l Conf. on Learning Representations. Vancouver: ICLR, 2018.

[43] Zhao PX. Research on adversarial attack method of time series prediction based on local perturbation [MS. Thesis]. Beijing: Beijing Jiaotong University, 2022 (in Chinese with English abstract). [doi: 10.26944/d.cnki.gbfju.2022.002751]

附中文参考文献:

[1] 原继东, 王志海. 时间序列的表示与分类算法综述. 计算机科学, 2015, 42(3): 1–7. [doi: 10.11896/j.issn.1002-137X.2015.3.001]

[6] 魏池璇, 王志海, 原继东, 林钱洪. 时间序列可变尺度的时频特征求解及其分类. 软件学报, 2022, 33(12): 4411–4428. http://www.jos.org.cn/1000-9825/6346.htm [doi: 10.13328/j.cnki.jos.006346]

[43] 赵培翔. 基于局部扰动的时间序列预测对抗攻击方法研究 [硕士学位论文]. 北京: 北京交通大学, 2022. [doi: 10.26944/d.cnki.gbfju.2022.002751]

张耀元(2000－), 女, 硕士生, 主要研究领域为时间序列对抗攻击.

王志海(1963－), 男, 博士, 教授, 博士生导师, CCF 专业会员, 主要研究领域为数据挖掘, 时间序列.

原继东(1989－), 男, 博士, 副教授, CCF 专业会员, 主要研究领域为数据挖掘, 时间序列分类.

赵培翔(1998－), 男, 硕士生, 主要研究领域为时间序列对抗攻击.

刘海洋(1987－), 男, 博士, 讲师, CCF 专业会员, 主要研究领域为数据挖掘, 人工智能.