基于局部影响分析模型的图神经网络对抗攻击

吴翼腾[①] 刘 伟[①] 于洪涛*[①] 操晓春[②]

[①](信息工程大学 郑州 450002)

[②](中国科学院信息工程研究所信息安全国家重点实验室 北京 100093)

摘 要：图神经网络(GNN)容易受到对抗攻击安全威胁。现有研究未注意到图神经网络对抗攻击与统计学经典分支统计诊断之间的联系。该文分析了二者理论本质的一致性，将统计诊断的重要成果局部影响分析模型引入图神经网络对抗攻击。首先建立局部影响分析模型，提出并证明针对图神经网络攻击的扰动筛选公式，得出该式的物理意义为扰动对模型训练参数影响的度量。其次为降低计算复杂度，根据扰动筛选公式的物理意义得出扰动筛选近似公式。最后引入投影梯度下降算法实施扰动筛选。实验结果表明，将局部影响分析模型引入图神经网络对抗攻击领域具有合理性；与现有攻击方法相比，所提方法具有有效性。

关键词：图神经网络；对抗攻击；统计诊断；局部影响分析；投影梯度下降

中图分类号：TN915.08; TP18 **文献标识码**：A **文章编号**：1009-5896(2022)07-2576-08

DOI: 10.11999/JEIT210448

Adversarial Attacks on Graph Neural Network Based on Local Influence Analysis Model

WU Yiteng[①] LIU Wei[①] YU Hongtao[①] CAO Xiaochun[②]

[①](*Information Engineering University*, *Zhengzhou* 450002, *China*)

[②](*State Key Laboratory of Information Security*, *Institute of Information Engineering*, *Chinese Academy of Sciences*, *Beijing* 100093, *China*)

Abstract: Graph Neural Networks (GNNs) are vulnerable to adversarial attacks. Existing papers do not pay attention to the relationship between adversarial attacks and statistical diagnosis, a classical branch of statistics. In this paper, the consistency of the two theories is analyzed, and the local influence analysis model, an important achievement of statistical diagnosis, is introduced into adversarial attack on GNNs. Firstly, the local influence analysis model is established to derive the equation of perturbation selecting of attacks, and the physical meaning of this equation is a measurement of the influence of perturbation on model training parameters. Secondly, to reduce the computational complexity, according to the physical meaning of the perturbation selecting equation, the approximate equation is obtained. Finally, the projected gradient descent algorithm is introduced to implement disturbance selecting. Experimental results show that it is reasonable to introduce the local influence analysis model into the field of adversarial attacks on graph neural network; Compared with the existing attack methods, the proposed method is more effective.

Key words: Graph Neural Network (GNN); Adversarial attack; Statistical diagnosis; Local influence analysis; Projected gradient descent

收稿日期：2021-05-25；改回日期：2021-12-21；网络出版：2022-02-03

*通信作者： 于洪涛 yht_ndsc@126.com

基金项目：自然科学基金创新研究群体项目(61521003)，国家重点研发计划(2016QY03D0502)， 郑州市协同创新重大专项基金(162/32410218)

Foundation Items: The Innovative Research Groups of the National Natural Science Foundation of China (61521003), The National Key R&D Project (2016QY03D0502), Zhengzhou City Collaborative Innovation Major Project (162/32410218)

1　引言

图结构数据和节点的语义属性数据有效结合，可以自然而完整地表达现实数据。类似于图数据库区别于传统关系型数据库，在存储个体信息的同时，还有效建立了个体之间的联系。图神经网络也区别于经典深度学习模型和传统复杂网络模型，它的主要特色是可以对语义属性数据和图数据统一表达建模，解决了传统研究中语义属性数据和图数据割裂的问题。图神经网络不仅在链路预测、节点分类等复杂网络任务，还在语义解析、视觉问答等自然语言处理任务和计算机视觉任务中展现了令人信服的性能[1,2]，是一类极具竞争力的深度学习模型[3]。

最新研究表明备受关注的图神经网络容易受到对抗攻击安全威胁[4]。2018年Zügner等人[5]首次提出图神经网络(Graph Neural Network, GNN)的对抗攻击。图神经网络的对抗攻击研究逐渐活跃于Conference and Workshop on Neural Information Processing Systems[6], International World Wide Web Conference[7], International Conference on Machine Learning[8]等顶级学术会议。

本文认为，与对抗攻击直接关联的研究应该始于20世纪70年代的统计诊断[9–11]。统计诊断系统研究了实际数据与既定模型之间的偏离，用于分析输入数据扰动对统计推断的影响[12]。对抗攻击指有目的地对输入数据施加微小扰动，使模型输出错误的预测结果[13,14]。可见，对抗攻击和统计诊断的理论本质相同，同属于模型安全问题的攻防两方面研究。

然而现有研究未注意到对抗攻击与统计诊断之间的联系，得出与统计诊断经典成果相似或相同的研究结论。例如图神经网络对抗攻击的经典文献[15]，根据攻击方法，作者从实验中得出有效攻击的原因是扰动后的数据诱导图神经网络训练出不良参数的结论。文献[16]系统研究了图神经网络对抗攻击的参数差异假设，认为扰动前后图神经网络的训练参数差异是形成有效攻击的重要因素之一。这些针对对抗攻击的研究结论与统计诊断中局部影响分析模型的基本假设一致。

局部影响分析是统计诊断最著名的研究成果之一，是Cook首先提出的一种很一般的统计诊断方法，适用于各种统计模型[11,12]。局部影响分析的基本假设是：对于异常点或强影响点，输入数据的微小扰动会对模型训练参数带来较大影响[17]。鉴于文献[15]“攻击后的图神经网络得到的训练参数较差”的实证研究结论、文献[16]提出的“参数差异是实施有效攻击的重要机理”，与局部影响分析模型的基本假设相吻合，本文考虑将局部影响分析模型引入图神经网络对抗攻击。与统计诊断的研究目标相反，选择扰动影响大的数据点实施攻击，使得重训练的图神经网络模型输出错误预测结果。

图神经网络对抗攻击领域引入局部影响分析有以下难点：一是需针对图神经网络场景重新推导扰动筛选公式；二是直接使用局部影响分析模型的主对角元法[12]实施扰动筛选计算量大，需对公式进一步简化，并采用更加有效的扰动筛选算法。本文的主要工作如下：

(1) 将统计诊断的局部影响分析模型引入图神经网络对抗攻击。针对目标图神经网络模型结构，推导出局部影响分析模型中参数差异度量公式。为降低主对角元法的时间和空间复杂度，得出基于攻击梯度的扰动筛选近似公式。

(2) 引入文献[18]中图神经网络对抗攻击行之有效的投影梯度下降 (Projected Gradient Descent, PGD)算法更新扰动。并通过实验验证了所提攻击方法的有效性，进一步说明了局部影响分析模型的合理性。

2　图神经网络和对抗攻击

图表示为$G(V,E)$，其中V表示节点集合，E表示连边集合。设节点数$|V|=N$，则无权无向图可用对称的邻接矩阵$\boldsymbol{A}=\{0,1\}^{N\times N}$表示，$\boldsymbol{A}^{\mathrm{T}}=\boldsymbol{A}$。图中每个节点有$n$维的特征向量，节点特征用矩阵$\boldsymbol{X}=\{0,1\}^{N\times n}$表示。文献[19–21]将图神经网络简化为SGC(Simple Graph Convolution)，它具有低通滤波器的作用。本文以SGC为攻击和研究对象。它可以表达为

$$\underset{N\times m}{\hat{\boldsymbol{Y}}}=\mathrm{softmax}(\underset{N\times N}{\mathcal{A}}\underset{N\times n}{\boldsymbol{X}}\underset{n\times m}{\boldsymbol{W}}) \tag{1}$$

其中，$\mathcal{A}$为滤波矩阵，$\boldsymbol{X}$为输入特征向量，$\boldsymbol{W}$为参数矩阵，$\hat{\boldsymbol{Y}}$为SGC模型的输出。在文献[21]中$\mathcal{A}$的形式通常为

$$\mathcal{A}=\tilde{\boldsymbol{L}}^{k},\tilde{\boldsymbol{L}}=\boldsymbol{D}^{-\frac{1}{2}}\tilde{\boldsymbol{A}}\boldsymbol{D}^{-\frac{1}{2}},\tilde{\boldsymbol{A}}=\boldsymbol{I}+\boldsymbol{A},\tilde{\boldsymbol{D}}=\mathrm{diag}(\tilde{\boldsymbol{A}}_1) \tag{2}$$

设$\boldsymbol{Z}=\mathcal{A}\boldsymbol{X}$，$\boldsymbol{Y}$为标签矩阵，使用交叉熵损失函数，可得到矩阵形式的损失函数计算公式

$$\mathcal{L}(\boldsymbol{W};\boldsymbol{A})=-\mathrm{tr}[\boldsymbol{Y}^{\mathrm{T}}\ln[\mathrm{softmax}(\boldsymbol{Z}\boldsymbol{W})]] \tag{3}$$

本文研究非指定目标、数据投毒攻击，并基于增删连边的扰动策略建立局部影响分析模型。非指定目标攻击不指定具体的1个或几个攻击目标，需要使测试集的准确率整体下降；投毒攻击指允许图卷积网络对污染的训练数据重新训练，重训练的图卷积网络在测试集的准确率仍然下降。文献[15, 18]建立了图神经网络非指定目标投毒攻击模型。投毒

攻击通常分为对抗训练和投毒攻击两个过程。在对抗训练阶段，图神经网络基于当前扰动数据进行正向训练。在投毒攻击阶段，图神经网络基于训练好的模型实施攻击，因此投毒攻击属于双层优化问题。

3 图神经网络对抗攻击的局部影响分析模型

本节建立图神经网络对抗攻击的局部影响分析模型。首先推导出局部影响分析模型的扰动筛选公式；然后根据扰动筛选公式的物理意义和表达形式对其简化；最后从文献[18]中引入投影梯度下降算法实施扰动筛选。

3.1 局部影响分析的扰动筛选公式

局部影响分析模型的主要思想是，训练数据扰动后损失函数会发生改变，通过比较扰动前后损失函数之间的差异进行扰动筛选。通过研究文献[11, 12, 22]可以得出结论，局部影响分析模型的核心是将模型参数视作扰动的函数，而不是将参数视为与扰动无关的、独立于扰动之外的变量。这将应用于损失距离的推导，是实施有效攻击的关键。

首先定义损失距离，用以衡量扰动后与扰动前损失函数的变化量。

定义 (损失距离)

$$\mathrm{LD}(\hat{\boldsymbol{A}}) = 2[\mathcal{L}(\boldsymbol{W}^*(\hat{\boldsymbol{A}})) - \mathcal{L}(\boldsymbol{W}_0)] \tag{4}$$

其中，$\boldsymbol{W}_0 = \boldsymbol{W}^*(\boldsymbol{A})$，$\mathcal{L}(\boldsymbol{W}_0) = \mathcal{L}(\boldsymbol{W}^*(\boldsymbol{A}))$。由于$\mathcal{L}(\boldsymbol{W}_0)$为全局最小值，因此恒有$\mathrm{LD} \geq 0$。

由以上定义可知，损失距离$\mathrm{LD}(\hat{\boldsymbol{A}})$越大，扰动$\hat{\boldsymbol{A}}$对参数$\boldsymbol{W}^*$估计的影响越大。为得到损失距离，直接根据定义计算通常比较复杂：需要对所有的可能扰动逐一遍历，然后重训练图神经网络并计算损失距离。然而对抗攻击问题只需在所有可能的扰动中，筛选出扰动相对影响最大的元素实施扰动。因此只需得到损失距离较好的近似公式即可。以下定理得出了损失距离的2阶近似公式，它是局部影响分析模型的核心公式。

设图神经网络式(1)采用梯度下降法训练

$$\boldsymbol{W}^t = \boldsymbol{W}^{t-1} - \alpha \cdot \nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W}^{t-1}) \tag{5}$$

则经过$t = t_0$轮训练，可得模型的训练参数$\boldsymbol{W}^* = \boldsymbol{W}^{t_0}$。推导中假定投毒后的训练数据构成的损失函数$\mathcal{L}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))$存在2阶以上连续偏导数。这时参数估计$\boldsymbol{W}^*(\hat{\boldsymbol{A}})$满足方程

$$\mathrm{vec}[\nabla_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))] = \left.\frac{\partial \mathcal{L}(\boldsymbol{W})}{\partial \mathrm{vec}(\boldsymbol{W})}\right|_{\boldsymbol{W}=\boldsymbol{W}^*(\hat{\boldsymbol{A}})} = 0, \forall \hat{\boldsymbol{A}} \tag{6}$$

其中，$\mathrm{vec}(\cdot)$表示矩阵按列优先拉直为向量。$\nabla_{\boldsymbol{W}}\mathcal{L}$表示损失函数$\mathcal{L}$对参数矩阵$\boldsymbol{W}$求梯度得到的与$\boldsymbol{W}$同型的矩阵。

定理 (损失距离的2阶近似) $\mathrm{LD}(\hat{\boldsymbol{A}}) = 2[\mathcal{L}(\boldsymbol{W}^*(\hat{\boldsymbol{A}})) - \mathcal{L}(\boldsymbol{W}_0)]$的2阶近似可以表示为

$$\mathrm{LD}^{\mathrm{II}}(\hat{\boldsymbol{A}}) = \underset{1\times NN}{\mathrm{vec}^{\mathrm{T}}(\boldsymbol{D})} \underset{NN\times NN}{\boldsymbol{F}} \underset{NN\times 1}{\mathrm{vec}(\boldsymbol{D})} \tag{7}$$

$$\underset{N\times N}{\boldsymbol{D}} = \hat{\boldsymbol{A}} - \boldsymbol{A} \tag{8}$$

$$\underset{NN\times NN}{\boldsymbol{F}} = \underset{NN\times nm}{\boldsymbol{G}} \underset{nm\times nm}{\nabla^2_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W}_0)} \underset{nm\times NN}{\boldsymbol{G}^{\mathrm{T}}} \tag{9}$$

$$\underset{NN\times nm}{\boldsymbol{G}} = \left.\frac{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})}\right|_{\hat{\boldsymbol{A}}=\boldsymbol{A}} \tag{10}$$

其中，$\boldsymbol{F}$称为影响矩阵。

证明

$$\begin{aligned}\mathrm{LD}^{\mathrm{II}}(\hat{\boldsymbol{A}}) =& \mathrm{LD}(\boldsymbol{A}) + \mathrm{vec}^{\mathrm{T}}(\nabla_{\hat{\boldsymbol{A}}} \mathrm{LD}(\boldsymbol{A}))\mathrm{vec}(\boldsymbol{D}) \\ &+ \frac{1}{2}\mathrm{vec}^{\mathrm{T}}(\boldsymbol{D})\nabla^2_{\hat{\boldsymbol{A}}} \mathrm{LD}(\boldsymbol{A})\mathrm{vec}(\boldsymbol{D}) + R(\boldsymbol{D})\end{aligned}$$

$$\mathrm{LD}(\boldsymbol{A}) = 2[\mathcal{L}(\boldsymbol{W}^*(\boldsymbol{A})) - \mathcal{L}(\boldsymbol{W}_0)] = 0$$

$$\begin{aligned}\mathrm{vec}(\nabla_{\hat{\boldsymbol{A}}} \mathrm{LD}(\boldsymbol{A})) =& 2\frac{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})} \left.\frac{\partial \mathcal{L}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}\right|_{\hat{\boldsymbol{A}}=\boldsymbol{A}} \\ =& 0\end{aligned}$$

根据式(6)，$\left.\dfrac{\partial \mathcal{L}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}\right|_{\hat{\boldsymbol{A}}=\boldsymbol{A}} = 0$，因此

$$\mathrm{LD}^{\mathrm{II}}(\hat{\boldsymbol{A}}) \approx \frac{1}{2}\mathrm{vec}^{\mathrm{T}}(\boldsymbol{D})\nabla^2_{\hat{\boldsymbol{A}}} \mathrm{LD}(\boldsymbol{A})\mathrm{vec}(\boldsymbol{D})$$

$$\begin{aligned}\frac{1}{2}\nabla^2_{\hat{\boldsymbol{A}}} \mathrm{LD}(\hat{\boldsymbol{A}}) =& \nabla^2_{\hat{\boldsymbol{A}}} \mathcal{L}(\boldsymbol{W}^*(\hat{\boldsymbol{A}})) = \frac{\partial \mathrm{vec}(\nabla_{\hat{\boldsymbol{A}}} \mathcal{L}(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})} \\ =& 2\frac{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})} \frac{\partial \mathrm{vec}(\nabla_{\hat{\boldsymbol{A}}} \mathcal{L}(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))} \\ =& \frac{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})} \\ &\cdot \frac{\partial \mathrm{vec}\left(\dfrac{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})} \dfrac{\partial \mathcal{L}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}\right)}{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))} \\ =& \frac{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})} \left\{ \frac{\partial \mathrm{vec}\left(\dfrac{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})}\right)}{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))} \right. \\ &\cdot \left[\frac{\partial \mathcal{L}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))} \otimes \boldsymbol{I}\right] \\ &+ \frac{\partial \mathrm{vec}\left(\dfrac{\partial \mathcal{L}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}\right)}{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))} \\ &\left. \cdot \left[\boldsymbol{I}_1 \otimes \frac{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})}\right] \right\}\end{aligned}$$

$$=\frac{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})}\frac{\partial \mathrm{vec}\left(\dfrac{\partial \mathcal{L}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}\right)}{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}\cdot\left[\frac{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})}\right]^{\mathrm{T}}$$

所以，

$$\frac{1}{2}\nabla^2_{\hat{\boldsymbol{A}}}\mathrm{LD}(\hat{\boldsymbol{A}})\Big|_{\hat{\boldsymbol{A}}=\boldsymbol{A}}=\boldsymbol{G}\nabla^2_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{W}_0)\boldsymbol{G}^{\mathrm{T}}=\boldsymbol{F}\quad\text{证毕}$$

定理表明，原本需要对邻接矩阵$\boldsymbol{A}$的$N\times N$个元素逐一扰动得到$\hat{\boldsymbol{A}}$，对$N\times N$次扰动使用式(5)重新训练图神经网络式(1)，并根据式(4)计算损失距离LD。现不需要对图神经网络重训练，而只需针对$N\times N$个扰动计算$N\times N$次矩阵乘法如式(7)。由于扰动类型为增删连边，根据式(8)，向量$\mathrm{vec}(\boldsymbol{D})$第$i$个元素为1或者–1，其他元素均为0而没有扰动。损失距离的2阶近似式(7)$\mathrm{LD}^{\mathrm{II}}(\hat{\boldsymbol{A}})$反映了第$i$个分量扰动对于损失函数的影响，其值越大扰动影响也越大。因此可选择扰动影响较大的扰动点实施攻击。

对于$\mathrm{LD}^{\mathrm{II}}(\hat{\boldsymbol{A}})$的计算，容易看出，若$\mathrm{vec}(\boldsymbol{D})$的第$i$个元素非零，其他元素为零，则$\mathrm{LD}^{\mathrm{II}}(\hat{\boldsymbol{A}})$的值等于影响矩阵$\boldsymbol{F}$的第$i$个主对角元素。因此可直接取出影响矩阵$\boldsymbol{F}$的主对角元素进行扰动筛选。即局部影响分析模型中扰动筛选的经典方法，主对角元法[12]。

3.2 扰动筛选公式的近似

根据3.1节分析，基于局部影响分析的对抗攻击方法主要关心影响矩阵$\boldsymbol{F}$的主对角元素，而与其他元素无关。若能直接计算$\boldsymbol{F}$的主对角元素而忽略其他元素，将会使时间和空间复杂度进一步降低为原来的$1/N$。

分析式(9)影响矩阵$\boldsymbol{F}$的物理意义。由于$\boldsymbol{F}=\boldsymbol{G}\nabla^2_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{W}_0)\boldsymbol{G}^{\mathrm{T}}$，而$\boldsymbol{G}$表示参数$\boldsymbol{W}^*(\hat{\boldsymbol{A}})$对$\mathrm{vec}(\hat{\boldsymbol{A}})$各个元素的偏导数，刻画了$\mathrm{vec}(\hat{\boldsymbol{A}})$各个元素扰动对参数矩阵$\boldsymbol{W}^*(\hat{\boldsymbol{A}})$的影响。而$\boldsymbol{F}$的第$i$个主对角元则表示$\mathrm{vec}(\hat{\boldsymbol{A}})$第$i$个元素扰动对参数$\boldsymbol{W}^*(\hat{\boldsymbol{A}})$各元素影响的加权和，加权矩阵为$\nabla^2_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{W}_0)$。

基于以上分析，局部影响分析的主要方法是计算参数对各个元素扰动的偏导数并进行扰动筛选。考虑将式(9)的主对角元法简化，直接根据$\boldsymbol{F}$的物理意义定义参数矩阵$\boldsymbol{W}^*(\hat{\boldsymbol{A}})$的范数$d$如式(11)所示，并计算$d$对$\hat{\boldsymbol{A}}$的各个元素的偏导数，得到攻击梯度矩阵

$$d=\sqrt{\mathrm{vec}^{\mathrm{T}}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))\nabla^2_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{W}_0)\mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}\quad(11)$$

事实上为便于计算，攻击梯度矩阵仅需考虑d的平方。攻击梯度矩阵$\nabla_{\hat{\boldsymbol{A}}}d^2$可表示为

$$\begin{aligned}\mathrm{vec}(\nabla_{\hat{\boldsymbol{A}}}d^2)=&\frac{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})}\frac{\partial d^2}{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}\\=&2\frac{\partial \mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))}{\partial \mathrm{vec}(\hat{\boldsymbol{A}})}\nabla^2_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{W}_0)\mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))\\=&2\boldsymbol{G}\nabla^2_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{W}_0)\mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))\end{aligned}\quad(12)$$

式(12)表明，攻击梯度矩阵$\nabla_{\hat{\boldsymbol{A}}}d^2$与$\boldsymbol{F}$的主对角元十分接近：式(12)包含了参数矩阵$\boldsymbol{W}^*(\hat{\boldsymbol{A}})$对$\hat{\boldsymbol{A}}$的偏导数矩阵$\boldsymbol{G}$的信息，以及加权矩阵$\nabla^2_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{W}_0)$的信息；区别在于$\boldsymbol{F}$的主对角元在权矩阵$\nabla^2_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{W}_0)$的两侧全部用$\boldsymbol{G}$加权求和，而攻击梯度$\nabla_{\hat{\boldsymbol{A}}}d^2$仅在左侧使用$\boldsymbol{G}$，而右侧统一使用参数的拉直向量$\mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))$加权($\mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))$的维数与矩阵$\boldsymbol{G}$中的每一列相同。由于针对每个扰动元素，右侧加权向量$\mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))$完全相同，因此各个扰动影响的区别主要体现在左侧的参数梯度矩阵$\boldsymbol{G}$和权矩阵$\nabla^2_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{W}_0)$之中。公式(12)的物理含义与$\boldsymbol{F}$的主对角元近似，但时间和空间复杂度却减少为原本计算$\boldsymbol{F}$的$1/N$。该复杂度与对抗攻击的经典方法Mettack[15]一致。

3.3 基于局部影响分析模型的对抗攻击算法

根据3.1节和3.2节的结论，基于局部影响分析的图神经网络对抗攻击模型可以表述为如下约束优化问题

$$\hat{\boldsymbol{A}}=\arg\max_{\hat{\boldsymbol{A}}}\left[\mathrm{vec}^{\mathrm{T}}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))\nabla^2_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{W}_0)\mathrm{vec}(\boldsymbol{W}^*(\hat{\boldsymbol{A}}))\right]\quad(13)$$

$$\text{s.t. }\boldsymbol{W}^*=\arg\min_{\boldsymbol{W}}\mathcal{L}(\boldsymbol{W};\hat{\boldsymbol{A}}),\left\|\hat{\boldsymbol{A}}-\boldsymbol{A}\right\|_0\le\delta\quad(14)$$

其中，$\|\cdot\|_0$表示矩阵中非零元素的个数，$\left\|\hat{\boldsymbol{A}}-\boldsymbol{A}\right\|_0\le\delta$表示扰动量约束。式(13)、式(14)表明对抗攻击问题属于双层优化问题，可分为式(14)的对抗训练和式(13)的投毒攻击两个阶段。在对抗训练阶段，采用式(5)的梯度下降法训练；在投毒攻击阶段，本文基于攻击梯度式(12)，引入文献[18]的投影梯度下降算法实施扰动筛选。投影梯度下降算法的核心步骤可以简要概括为：

(1) 将式(13)邻接矩阵的$\{0,1\}$离散约束优化问题松弛为闭区间$[0,1]$上的连续约束优化问题，松弛后的约束条件对应为元素扰动总和$\boldsymbol{1}^{\mathrm{T}}\mathrm{vec}(\hat{\boldsymbol{A}}'-\boldsymbol{A})\le\delta$，其中$\boldsymbol{1}$为$N^2$维全1列向量，$\hat{\boldsymbol{A}}'$为连续值扰动后的邻接矩阵。

(2) 采用解决简单连续约束优化问题的投影梯

度下降算法，先进行通常的梯度下降，再对更新的连续取值的$\hat{\boldsymbol{A}}'$矩阵作投影，以保证扰动总量满足$\boldsymbol{1}^{\mathrm{T}}\mathrm{vec}(\hat{\boldsymbol{A}}'-\boldsymbol{A})\leq\delta$的约束条件。

(3) 最后将扰动后连续取值的$\hat{\boldsymbol{A}}'$还原为$\{0,1\}$取值的邻接矩阵$\hat{\boldsymbol{A}}$。

根据以上分析，本文提出的基于局部影响分析模型的图神经网络对抗攻击算法如表1所示。

4 实验

实验采用型号为TITAN Xp的GPU显卡，运行环境为Ubuntu 16.04系统，Cuda10.0，Python3.6以及Pytorch1.4。实验采用Polblogs[23] , Cora_ml, Cora[24], Citeseer[25]等4个数据集，数据集的统计特性如表2所示。将提出的基于局部影响分析的图神经网络对抗攻击方法与随机攻击方法Random和经典的投毒攻击方法Mettack, Min-max进行对比实验。Mettack的攻击梯度采用approximating meta-gradients[15]，扰动筛选算法为贪心算法。Min-max的攻击梯度为CE[18]，扰动筛选算法采用投影梯度下降算法，投影梯度下降的学习率设置为$100/\sqrt{t}$(t为当前迭代轮数，取迭代总轮数为100)。本文所提方法的攻击梯度采用式(12)计算，投影梯度下降算法的学习率设置与Min-max相同。对于Polblogs, Cora_ml数据集，图神经网络SGC的正向训练的学习率取α=0.1；对于Cora, Citeseer数据集，学习率取α=0.01。允许攻击方法对训练集中连边总数的3%进行攻击。实验中划分40%训练集和60%的测试集，数据集随机划分10次，记录10次SGC的初始准确率和攻击后准确率的平均值，得到表3的实验数据。表中准确率的最小值用黑体标出。

分析表中的实验数据，可以得出以下结论：

表 1 基于局部影响分析模型的图神经网络对抗攻击算法

输入：邻接矩阵$\boldsymbol{A}$，特征矩阵$\boldsymbol{X}$，标签$\boldsymbol{Y}$，攻击点数n，迭代次数iters；

输出：扰动列表disturb_list

(1) disturb_list = [ ];

(2) for i = 1:iters:

(3) 根据disturb_list更新扰动矩阵$\hat{\boldsymbol{A}}'$；

(4) 根据式(14)和式(5)重训练图神经网络得参数$\boldsymbol{W}^*$；

(5) 根据式(12)计算攻击梯度$\nabla_{\hat{\boldsymbol{A}}}d^2$；

(6) 根据攻击梯度$\nabla_{\hat{\boldsymbol{A}}}d^2$采用梯度下降算法更新扰动矩阵$\hat{\boldsymbol{A}}'$；

(7) 对$\hat{\boldsymbol{A}}'$进行投影操作以控制扰动总量满足约束条件，disturb list=$\hat{\boldsymbol{A}}'-\boldsymbol{A}$；

(8) end for

(9) 把$\hat{\boldsymbol{A}}'$还原为邻接矩阵$\hat{\boldsymbol{A}}$；

(10) 返回disturb list=$\hat{\boldsymbol{A}}-\boldsymbol{A}$。

(1) 采用随机增删连边的方式Random无法实现有效的投毒攻击。有效的投毒攻击需要针对模型结构或训练方法设计加扰方式。

(2) 虽然$k=1$时图神经网络SGC未受扰动时的模型预测准确率相比$k=2$时低，但是模型的鲁棒性更高，在同样的扰动比例下$k=2$时更易受到对抗攻击威胁。原因可以从SGC模型公式(1)中分析得出：$k=2$时，由于需要计算矩阵$\hat{\boldsymbol{L}}$的平方，相当于将扰动影响进一步放大，模型更加脆弱。

(3) 经典攻击方法Mettack相比Min-max有更好的攻击效果。原因主要是Mettack采用的攻击梯度求解思想与本文基于局部影响分析模型的攻击梯度求解思想类似，将参数视为扰动的函数而非独立变量，Min-max在每轮攻击中将参数视为固定常数。将参数视为扰动的函数这一观点早已在统计诊断中发展并实践，这也是本文将对抗攻击研究溯源至统计诊断的原因之一。

(4) 基于局部影响分析模型的攻击方法能有效提高攻击性能。当$k=1$时攻击效果相比经典方法有1%左右的提升；$k=2$时相比经典方法攻击效果提升为2%～5%。将本文所提方法与Min-max相比，控制扰动筛选算法采用投影梯度下降算法不变，区别为攻击梯度的求解方式，本文采用局部影响分析模型的式(12)求解攻击梯度。实验结果表明本文所提方法更具有效性，实验结果支持了局部影响分析模型引入图神经网络对抗攻击的合理性。

表 2 数据集统计特性

数据集	节点数	连边数	特征维数	分类数
Polblogs	1222	16714	1490	2
Cora_ml	2810	7981	2879	7
Cora	2485	5069	1433	7
Citeseer	2110	3668	3703	6

表 3 本文方法与其他攻击方法的对比(%)

	方法	Polblogs	Cora_ml	Cora	Citeseer
$k=1$	未扰动	94.70	86.51	85.09	74.70
	Random	94.62	86.51	85.12	74.15
	Mettack	92.15	85.33	84.43	74.83
	Min-max	92.56	85.36	83.66	74.11
	本文方法	**91.37**	**84.87**	**83.55**	**73.12**
$k=2$	未扰动	95.65	88.02	87.24	74.72
	Random	95.62	88.11	87.20	74.48
	Mettack	94.06	80.52	80.94	73.02
	Min-max	95.05	87.65	86.49	75.11
	本文方法	**92.91**	**75.53**	**78.75**	**68.12**

为进一步比较不同方法的实验结果，说明不同扰动量对攻击效果的影响，其他实验条件不变，采用1%～5%的扰动并记录准确率下降的平均值，并绘制曲线如图1所示。

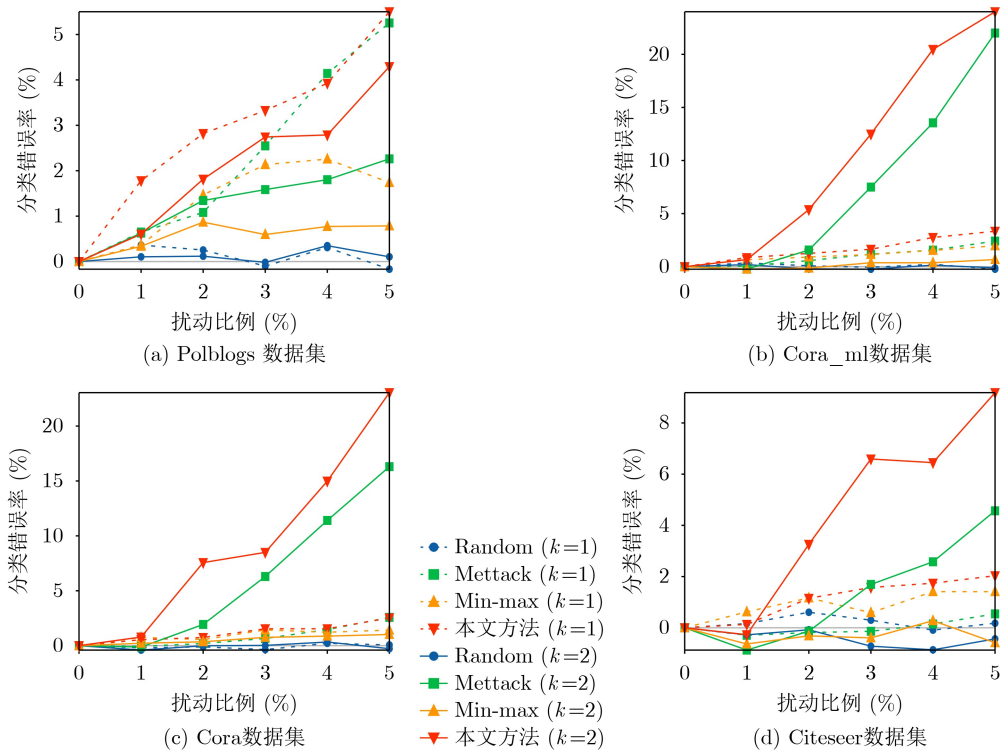

图 1 不同扰动量的攻击效果对比

总体而言，本文所提攻击方法具有更好的攻击效果，在4个数据集上对于不同k值几乎均超过经典方法Mettack和Min-max。实验结果支持了本文引入局部影响分析模型的合理性、本文所提攻击方法的有效性。

为进一步验证本文所提攻击方法对不同图神经网络的可扩展性，采用上述实验($k = 2$时)中各个攻击方法生成的投毒训练数据，作为图卷积神经网络GCN(Graph Convolutional Network)[26]和网络表示学习方法DeepWalk[27]+多类逻辑回归分类器的输入，对比两种方法在节点分类任务上的准确率。基于3%的扰动，保持其他实验条件不变，得到表4的实验数据。

实验结果表明，对于与本文直接攻击的目标图神经网络SGC原理相近的GCN模型，攻击效果具有较好的扩展性，本文所提的基于局部影响分析模型的攻击方法均取得较理想的攻击效果。对于网络表示学习方法DeepWalk，基于SGC模型得到的投毒数据在Polblogs数据集和Cora数据集的攻击效果较好；而在Cora_ml和Citeseer数据集上的攻击表现一般。一方面，由于DeepWalk获得节点向量表示的原理与SGC或GCN模型存在较大差异，投毒数据的迁移性减弱。另一方面，DeepWalk在节点分类任务的初始准确率普遍较低，尤其在Cora_ml，Cora和Citeseer数据集明显低于SGC或GCN模型，因此基于较高初始准确率获得的投毒扰动容易被较低的初始预测准确率淹没，从而不易体现投毒数据的攻击性能。

5　结束语

因图神经网络应用广泛，其安全问题备受关注。本文关注到对抗攻击与统计学经典分支统计诊断的联系，把统计诊断的重要成果局部影响分析模

表 4　投毒数据用于攻击其他图学习模型

	方法	Polblogs	Cora_ml	Cora	Citeseer
GCN	未扰动	94.57%	87.12%	85.20%	74.93%
	Random	94.34%	87.26%	85.13%	74.72%
	Mettack	92.45%	80.59%	81.02%	74.27%
	Min-max	92.77%	85.16%	84.83%	74.01%
	本文方法	**91.22%**	**78.49%**	**79.71%**	**70.36%**
DeepWalk	未扰动	92.23%	79.32%	74.29%	58.35%
	Random	92.19%	80.03%	74.44%	59.13%
	Mettack	91.06%	**76.63%**	72.27%	60.24%
	Min-max	92.15%	78.78%	73.75%	**57.37%**
	本文方法	**90.36%**	77.45%	**71.47%**	58.93%

型引入图神经网络对抗攻击。推导出局部影响分析模型的扰动筛选公式——损失距离的2阶近似。该式的物理意义是扰动对模型训练参数的影响度量。结合损失距离2阶近似公式的物理意义和其表达形式，定义攻击梯度作为扰动筛选的近似公式，使模型复杂度降低为原来的$1/N$。尔后引入投影梯度下降算法进行扰动筛选，并设计了基于局部影响分析模型的完整攻击算法。实验结果支持了局部影响分析模型的合理性和攻击方法的有效性。

局部影响分析模型是一类较广泛的扰动分析模型，不仅适用于本文的增删连边攻击，也适用于特征扰动、节点注入、标签翻转等其他攻击类型，后续工作可针对不同攻击类型作进一步推广。局部影响分析模型还可扩展至图像、文本等其他数据类型的对抗样本研究。

参 考 文 献

[1] 白铂, 刘玉婷, 马驰骋, 等. 图神经网络[J]. 中国科学:数学, 2020, 50(3): 367–384. doi: 10.1360/N012019-00133.
BAI Bo, LIU Yuting, MA Chicheng, *et al*. Graph neural network[J]. *Scientia Sinica*:*Mathematica*, 2020, 50(3): 367–384. doi: 10.1360/N012019-00133.

[2] 康世泽, 吉立新, 张建朋. 一种基于图注意力网络的异质信息网络表示学习框架[J]. 电子与信息学报, 2021, 43(4): 915–922. doi: 10.11999/JEIT200034.
KANG Shize, JI Lixin, and ZHANG Jianpeng. Heterogeneous information network representation learning framework based on graph attention network[J]. *Journal of Electronics & Information Technology*, 2021, 43(4): 915–922. doi: 10.11999/JEIT200034.

[3] 徐冰冰, 岑科廷, 黄俊杰, 等. 图卷积神经网络综述[J]. 计算机学报, 2020, 43(5): 755–780. doi: 10.11897/SP.J.1016.2020.00755.
XU Bingbing, CEN Keting, HUANG Junjie, *et al*. A survey on graph convolutional neural network[J]. *Chinese Journal of Computers*, 2020, 43(5): 755–780. doi: 10.11897/SP.J.1016.2020.00755.

[4] XU Han, MA Yao, LIU Haochen, *et al*. Adversarial attacks and defenses in images, graphs and text: A review[J]. *International Journal of Automation and Computing*, 2020, 17(2): 151–178. doi: 10.1007/s11633-019-1211-x.

[5] ZÜGNER D, AKBARNEJAD A, and GÜNNEMANN S. Adversarial attacks on neural networks for graph data[C]. The 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, United Kingdom, 2018: 2847–2856. doi: 10.1145/3219819.3220078.

[6] MA Jiaqi, DING Shuangrui, and MEI Qiaozhu. Towards more practical adversarial attacks on graph neural networks[C]. The 34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada, 2020.

[7] LI Jia, ZHANG Honglei, HAN Zhichao, *et al*. Adversarial attack on community detection by hiding individuals[C]. The Web Conference 2020, Taipei, China, 2020: 917–927. doi: 10.1145/3366423.3380171.

[8] BOJCHEVSKI A and GÜNNEMANN S. Adversarial attacks on node embeddings via graph poisoning[C]. The 36th International Conference on Machine Learning, Long Beach, USA, 2019: 695–704.

[9] COOK R D. Detection of influential observation in linear regression[J]. *Technometrics*, 1977, 19(1): 15–18. doi: 10.1080/00401706.1977.10489493.

[10] COOK R D. Influential observations in linear regression[J]. *Journal of the American Statistical Association*, 1979, 74(365): 169–174. doi: 10.1080/01621459.1979.10481634.

[11] 韦博成, 鲁国斌, 史建清. 统计诊断引论[M]. 南京: 东南大学出版社, 1991: 442–488.
WEI Bocheng, LU Guobin, and SHI Jianqing. Introduction to Statistical Diagnosis[M]. Nanjing: Southeast University Press, 1991: 442–488.

[12] 韦博成, 林金官, 解锋昌. 统计诊断[M]. 北京: 高等教育出版社, 2009: 101–118.
WEI Bocheng, LIN Jinguan, and XIE Fengchang. Statistical Diagnosis[M]. Beijing: Higher Education Press, 2009: 101–118.

[13] YUAN Xiaoyong, HE Pan, ZHU Qile, *et al*. Adversarial examples: Attacks and defenses for deep learning[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2019, 30(9): 2805–2824. doi: 10.1109/TNNLS.2018.2886017.

[14] 闫佳, 闫佳, 聂楚江, 等. 基于遗传算法的恶意代码对抗样本生成方法[J]. 电子与信息学报, 2020, 42(9): 2126–2133. doi: 10.11999/JEIT191059.
YAN Jia, YAN Jia, NIE Chujiang, *et al*. Method for generating malicious code adversarial samples based on genetic algorithm[J]. *Journal of Electronics & Information Technology*, 2020, 42(9): 2126–2133. doi: 10.11999/JEIT191059.

[15] ZÜGNER D and GÜNNEMANN S. Adversarial attacks on graph neural networks via meta learning[C]. The 7th International Conference on Learning Representations, New Orleans, USA, 2019.

[16] WU Yiteng, LIU Wei, HU Xinbang, *et al*. Parameter discrepancy hypothesis: Adversarial attack for graph data[J]. *Information Sciences*, 2021, 577: 234–244. doi: 10.1016/j.ins.2021.06.086.

[17] COOK R D and WEISBERG S. Residuals and Influence in Regression[M]. New York: Chapman and Hall, 1982: 1–20.

[18] XU Kaidi, CHEN Hongge, LIU Sijia, *et al*. Topology attack and defense for graph neural networks: An optimization

perspective[C]. The Twenty-Eighth International Joint Conference on Artificial Intelligence, Macao, China, 2019: 3961–3967. doi: 10.24963/ijcai.2019/550.

[19] LI Qimai, WU Xiaoming, LIU Han, *et al*. Label efficient semi-supervised learning via graph filtering[C]. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, USA, 2019: 9582–9591. doi: 10.1109/CVPR.2019.00981.

[20] NT H and MAEHARA T. Revisiting graph neural networks: All we have is low-pass filters[J]. arXiv: 1905.09550, 2019.

[21] WU F, ZHANG Tianyi, DE SOUZA JR A H, *et al*. Simplifying graph convolutional networks[J]. arXiv: 1902.07153, 2019.

[22] 费宇, 陈飞, 喻达磊, 等. 线性和广义线性混合模型及其统计诊断[M]. 北京: 科学出版社, 2013: 51–82.
FEI Yu, CHEN Fei, YU Dalei, *et al*. Linear and Generalized Linear Mixed Models and Their Statistical Diagnosis[M]. Beijing: Science Press, 2013: 51–82.

[23] SEN P, NAMATA G, BILGIC M, *et al*. Collective classification in network data[J]. *AI Magazine*, 2008, 29(3): 93–106. doi: 10.1609/aimag.v29i3.2157.

[24] MCCALLUM A K, NIGAM K, RENNIE J, *et al*. Automating the construction of internet portals with machine learning[J]. *Information Retrieval*, 2000, 3(2): 127–163. doi: 10.1023/A:1009953814988.

[25] ADAMIC L A and GLANCE N. The political blogosphere and the 2004 U. S. election: Divided they blog[C]. The 3rd International Workshop on Link Discovery, Chicago, USA, 2005: 36–43. doi: 10.1145/1134271.1134277.

[26] KIPF T N and WELLING M. Semi-supervised classification with graph convolutional networks[C]. The 5th International Conference on Learning Representations, Toulon, France, 2017.

[27] PEROZZI B, AL-RFOU R, and SKIENA S. Deepwalk: Online learning of social representations[C]. The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, USA, 2014: 701–710. doi: 10.1145/2623330.2623732.

吴翼腾：男，1992年生，博士，工程师，研究方向为人工智能安全、对抗机器学习.

刘　伟：男，1992年生，硕士，工程师，研究方向为人工智能安全、自然语言理解.

于洪涛：男，1970年生，博士，研究员，博士生导师，主要研究方向为人工智能与大数据.

责任编辑：马秀强