Bayesian Adversarial Attack on Graph Neural Networks (Student Abstract)

Xiao Liu, Jing Zhao, Shiliang Sun
School of Computer Science and Technology, East China Normal University,
3663 North Zhongshan Road, Shanghai 200241, P. R. China.
Email: jzhao, slsun@cs.ecnu.edu.cn

Abstract

Adversarial attack on graph neural network (GNN) is distinctive as it often jointly trains the available nodes to generate a graph as an adversarial example. Existing attacking approaches usually consider the case that all the training set is available which may be impractical. In this paper, we propose a novel Bayesian adversarial attack approach based on projected gradient descent optimization, called Bayesian PGD attack, which gets more general attack examples than deterministic attack approaches. The generated adversarial examples by our approach using the same partial dataset as deterministic attack approaches would make the GNN have higher misclassification rate on graph node classification. Specifically, in our approach, the edge perturbation Z is used for generating adversarial examples, which is viewed as a random variable with scale constraint, and the optimization target of the edge perturbation is to maximize the KL divergence between its true posterior distribution $p(Z|D)$ and its approximate variational distribution $q_\theta(Z)$. We experimentally find that the attack performance will decrease with the reduction of available nodes, and the effect of attack using different nodes varies greatly especially when the number of nodes is small. Through experimental comparison with the state-of-the-art attack approaches on GNNs, our approach is demonstrated to have better and robust attack performance.

Introduction

In this paper, we focus on the attack targeting the graph structure which is specific to graph neural networks (GNNs). The adversarial attack on the graph is to add edge perturbation on the original graph, i.e., adding or deleting certain edges, to construct a new graph as its adversarial example.

Adversarial attack on GNNs has two characteristics. First, the adversarial example is training set specific which is generated by jointly training a set of examples represented as nodes in the graph. This is largely different from other neural networks with non-graph structures in which the adversarial example is example specific. Second, attackers have to find adversarial perturbation in a discrete domain. The first characteristic has a deeper meaning that the adversarial examples with more nodes will get better attack performance. A common practice is to use all the training examples to generate adversarial examples (Xu et al. 2019). However, it may be unavailable to access the entire training set in real world. It is significant and has practical value to develop attack approaches in this situation. In this paper, we develop a novel Bayesian adversarial attack approach with better attack performance than the state-of-the-art approaches especially when there is partial data available. The second property requires converting the discrete edge disturbance to the continuous variable if the gradient-based method is used for optimization. Some work directly optimized the discrete perturbation variable (Xu et al. 2019). In our approach, we consider reparameterizing Z by the Gumbel-softmax trick.

Method

Let G be an unweigted and undirected graph. Let A denote the adjacency matrix of G. Under attack on graph structure, a new graph G' is constructed and its adjacency matrix is

$$A' = A \odot (1 - Z), \tag{1}$$

where $\odot$ is the element-wise product, $1 \in \{1\}^{N\times N}$ denotes a ones matrix and $Z \in \{0,1\}^{N\times N}$ denotes the edge perturbation. If $Z_{ij} = 0$, we keep the connection state between node i and node j; conversely, we change it (add an edge if $A_{ij} = 0$ or delete an edge if $A_{ij} = 1$).

In our attack approach, we define the random edge perturbation Z as

$$Z_{ij} = \begin{cases} 1 & with\ probability\ p_{ij}, \\ 0 & with\ probability\ 1 - p_{ij}. \end{cases} \tag{2}$$

Attackers try to find minimum edge perturbation Z to mislead GNNs. Our attack loss is

$$\begin{aligned} \theta^* &= \arg\max_\theta KL(q_\theta(Z)||p(Z|D)) \\ &= \arg\min_\theta H(q) + \mathbb{E}_{q_\theta(Z)} \log[p(Z)] + \mathbb{E}_{q_\theta(Z)}[\log p(D|Z)] \\ s.t.\ \ & \mathbb{E}Z \le \epsilon, \end{aligned} \tag{3}$$

where $p(Z)$ denotes a given prior distribution of Z, $\log p(D|Z)$ denotes the log-likelihood which is negative cross entropy for classification, ϵ is a hyperparameter and $H(\cdot)$ denotes entropy. We constrain $\mathbb{E}Z$ so that the perturbation is imperceptible.

Table 1: Test misclassification rates (%) under 5% perturbed edges. “Clean” denotes the unattacked model.

	Cora	Citeseer
Clean	18.2	28.9
Greedy	25.2	34.6
CE-PGD	28.0	36.0
Bayesian PGD (ours)	$\mathbf{28.2 \pm 0.2}$	$\mathbf{36.9 \pm 0.1}$

The unbiased Monte Carlo gradients of (3) is,

$$\frac{\partial}{\partial \theta}\mathbb{E}_{q_\theta(Z)} f(Z,\theta) = \mathbb{E}_{q(\xi)}[\frac{\partial f(Z,\theta)}{\partial Z}\frac{\partial Z}{\partial \theta} + \frac{\partial f(Z,\theta)}{\partial \theta}], \quad (4)$$

where $f(Z, \theta)$ denotes the attack loss (see (3)), $Z = t(\xi)$ is transformed using the reparameterization trick as

$$Z = \text{sigmoid}((\log(\text{sigmoid}(\theta)) + \xi)/\gamma), \quad (5)$$

where $\gamma \in \Re$ is a hyperparameter, ξ is drawn from Gumbel(0, 1) (Jang, Gu, and Poole 2016) and $\theta \in \Re^{N\times N}$ is a trainable parameter which denotes the unscaled probabilities in (2) ($p_{ij} = \text{sigmoid}(\theta_{ij})$).

We solve (3) by employing multistep projected gradient descent (PGD),

$$\begin{aligned} &\theta^{(t)} = \Pi(\theta^{(t-1)} - \eta_t g_t) \\ &\Pi(\theta) = \begin{cases} \theta & \mathbb{E}Z \leq \epsilon, \\ \text{sigmoid}^{-1}[\text{sigmoid}(\theta)\frac{\epsilon}{\mathbb{E}Z}] & \mathbb{E}Z > \epsilon, \end{cases} \end{aligned} \quad (6)$$

where t denotes the iteration step, g_t denotes the gradient of attack loss evaluated at $\theta^{(t-1)}$ (see (4)), η_t is the learning rate at iteration t and Π denotes the projection that make $\theta^{(t)}$ satisfy the constraint in (3).

The stability of random attack’s performance can be guaranteed.

Experiment

We evaluate our approach on Cora and Citeseer. We choose the graph convolutional network (Kipf and Welling 2017), a special form of GNN, as the model architecture.

Attack Performance Evaluation

We compare our algorithm with some related state-of-the-art approaches, including greedy attack which is a variant of meta-self attack (Zügner and Günnemann 2019) and CE-PGD (Xu et al. 2019) with cross entropy as objective.

In the first experiment, we use the whole training set to generate the adversarial example, and evaluate the attack performance on the test set. In Table 1, we present the test misclassification rate of different attack methods under edge perturbation. ϵ is set to be 5% of the total number of existing edges. The adversarial example generated by our algorithm is random, and so we record average misclassification rate and standard deviation (std).

In the second experiment, we further study the generalization performance of the adversarial example on graph data. We randomly select n nodes from the training set to generate attack examples for five times. We report the average misclassification rate and its standard deviation (std) on test set in Figure 1. We gradually reduce n and observe the attack performance of CE-PGD and Bayesian PGD under 5% perturbed edges on test set. We discover that the misclassification rate decreases as n falls. It can be regarded as the overfitting of attack models. Meanwhile, the attack perfomance becomes unstable if using different nodes especially when n falls, which is reflected by the increasing std. If we compare the two methods further, we can find that the proposed Bayesian PGD attack has better and more robust attack performance than CE-PGD, which are reflected by the higher misclassification rate and lower std. The advantage of the proposed method may be attributed to the Bayesian framework which introduces regularizations on the perturbation to moderated overfitting.

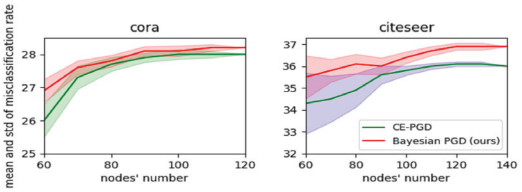

Figure 1: Mean and std of test misclassification rates.

Conclusion

In this paper, we propose a novel Bayesian attack approach, called *Bayesian PGD attack*, for adversarial attack on GNNs. Our approach considers the uncertainty of edge perturbation in the Bayesian framework and results in a random attack approach with high stability and generalization. Empirical results show that our algorithm can obtain more effective adversarial examples than deterministic methods. We believe that this paper provides a new perspective for the reseach on adversarial attack to GNNs.

Acknowledgments

This work is supported by the National NSFC Project 61673179, Shanghai Knowledge Service Platform Project ZF1213, and the special fund of ECNU for short-term postgraduate study abroad and international conference.

References

Jang, E.; Gu, S.; and Poole, B. 2016. Categorical reparameterization with Gumbel-softmax. *arXiv preprint arXiv:1611.01144,* 1–10.

Kipf, T. N., and Welling, M. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 1–14.

Xu, K.; Chen, H.; Liu, S.; Chen, P.-Y.; Weng, T.-W.; Hong, M.; and Lin, X. 2019. Topology attack and defense for graph neural networks: An optimization perspective. In *International Joint Conference on Artificial Intelligence*, 1–8.

Zügner, D., and Günnemann, S. 2019. Adversarial attacks on graph neural networks via meta learning. In *International Conference on Learning Representations*, 1–8.