Proof of Influence Analysis

USENIX Security '22 submission # 298: Fast Yet Effective Unlearning in Graph Neural Networks through Influence Analysis

1 Loss Function

In this paper, we consider the *cross-entropy* (CE) loss function as the loss function of GNNs. Specifically, given a graph $G(V,E)$ and a classifier, the CE loss of the classifier for any node $v \in V$ is defined as following

$$L(\theta; v, E) = -\sum_{c=1}^{\ell} y_{v,c} \log \frac{\exp(H[v,c])}{\sum_{k=1}^{\ell} \exp(H[v,k])}, \quad (1)$$

where $y_{v,c}$ denotes the TRUE/FALSE (0/1) ground truth of class c for node v, and $H \in \mathbb{R}^{|V| \times \ell}$ denotes the node embedding generated by AGGREGATE functions (defined by Eqn. (1) in the paper).

In this paper, we consider the *binary cross-entropy* (BCE) loss, i.e., $\ell = 2$ with two labels "1" and "2". Without losing generosity, we assume the true label of the given node v is label "1" (i.e., $y_{v,2} = 0$). Then we have the following:

$$L(\theta; v, E) = -\log \frac{\exp(H[v,1])}{\sum_{k=1}^{2} \exp(H[v,k])}. \quad (2)$$

Then, by dividing both numerator and denominator in (2) with $\exp(H[v,1])$, we get

$$\begin{aligned} L(\theta; v, E) &= -\log \frac{1}{1 + \exp(H[v,2]) / \exp(H[v,1])}. \\ &= \log\big(1 + \frac{\exp(H[v,2])}{\exp(H[v,1])}\big). \end{aligned} \quad (3)$$

It is well known that the log function can be approximated as $\log(1+x) \approx x$ when x is sufficiently small. Given a classifier with high accuracy, when the true label of the given node v is label "1", $\frac{\exp(H[v,2])}{\exp(H[v,1])} \to 0$. Therefore we have

$$L(\theta; v, E) \approx \exp(H[v,2] - H[v,1]). \quad (4)$$

If we apply the logarithm function to Eqn. (4) we can obtain the loss in a linear format:

$$\mathcal{L}(\theta; v, E) \approx H[v,2] - H[v,1]. \quad (5)$$

Following this, we define the BCE loss function in its logarithm format, and obtain the following loss function:

$$\mathcal{L}(\theta; v, E) = \log\Big[-\sum_{c=1}^{\ell} y_{v,c} \log \frac{\exp(H[v,c])}{\sum_{k=1}^{\ell} \exp(H[v,k])} \Big]. \quad (6)$$

Problem statement. Given a graph $G(V,E)$, a set of edges $E_{UL} \subset E$ that needs to be removed from G, and a GNN model θ, we will prove that the loss function (Eqn. 6) satisfies the following

$$\mathcal{L}(\theta; v, E) = \mathcal{L}(\theta; v, E \backslash E_{UL}) + \mathcal{L}(\theta; v, E_{UL}). \quad (7)$$

In the next subsections, we will prove Eqn. (7) holds for the three GNN models (GCN, GraphSAGE, and GIN) that we considered in the paper. Note that these GNN models have different aggregation functions to generate H in the loss function.

1.1 GCN

The typical aggregation method for GCN model is formulated as following:

$$H[v] = \text{AGGREGATE}_{GCN}(v) = \sum_{u \in N(v) \bigcup \{v\}} \frac{X[u]}{\sqrt{\hat{d}_u, \hat{d}_v}} \theta, \quad (8)$$

where $\hat{d}_u$ denotes the degree of node u plus 1, $N(v)$ is the set of nodes that connect to node v, $X \in \mathbb{R}^{N \times d}$ is a matrix represents node features, and $\theta \in \mathbb{R}^{\ell \times d}$ is the learnable parameter. We use $X_u \in \mathbb{R}^d$ to denote the node feature vector of node u.

To simplify our discussions, we omit the normalization term ($\sqrt{\hat{d}_u, \hat{d}_v}$) in Eqn. (8). Indeed our empirical results show that without normalization EraEdge still achieves competitive performance. In other words, we consider the GCN variant in which the aggregation function does not have normalization, i.e.,

$$H[v] = \sum_{u \in N(v) \bigcup \{v\}} X_u \theta^{\intercal}. \quad (9)$$

Without losing generosity, we assume the true label of the given node v is label "1" (i.e., $y_{v,2} = 0$). Then we obtain the following loss function of node v by a GCN model,

$$\mathcal{L}(\theta; v, E) \approx \sum_{u \in N(v) \cup \{v\}} X_u \theta[2] - \sum_{u \in N(v) \cup \{v\}} X_u \theta[1], \quad (10)$$

where $\theta[\cdot] \in \mathbb{R}^d$ denotes the trainable parameters for a label.

When we remove a set of edges E_{UL}, the loss of the node v by removing E_{UL} is formulated as

$$\mathcal{L}(\theta; v, E_{UL}) \approx \sum_{u \in \bar{N}(v)} X_u \theta[2] - \sum_{u \in \bar{N}(v)} X_u \theta[1], \quad (11)$$

where $\bar{N}(v) = \{u | (u, v) \in E_{UL}\}$.

Similarly, the loss of the node v by the remaining edges $E \backslash E_{UL}$ is defined as:

$$\mathcal{L}(\theta; v, E \backslash E_{UL}) \approx \sum_{u \in N(v) \backslash \bar{N}(v)} X_u \theta[2] - \sum_{u \in N(v) \backslash \bar{N}(v)} X_u \theta[1]. \quad (12)$$

Based on Eqns (10) - (12), we can have:

$$\begin{aligned} \mathcal{L}(\theta; v, E \backslash E_{UL}) + \mathcal{L}(\theta; v, E_{UL}) = & \Big(\sum_{u \in N(v) \backslash \bar{N}(v)} X_u \theta[2] + \sum_{u \in \bar{N}(v)} X_u \theta[2] \Big) \\ & - \Big(\sum_{u \in N(v) \backslash \bar{N}(v)} X_u \theta[1] + \sum_{u \in \bar{N}(v)} X_u \theta[1] \Big) \end{aligned} \quad (13)$$

It is easy to see that

$$(N(v) \backslash \bar{N}(v)) \cup \bar{N}(v) = N(v). \quad (14)$$

Therefore, we prove that

$$\mathcal{L}(\theta; v, E \backslash E_{UL}) + \mathcal{L}(\theta; v, E_{UL}) = \mathcal{L}(\theta; v, E).$$

1.2 GraphSAGE

The aggregation method of GraphSAGE can be formulated as

$$H_v = \mathsf{AGGREGATE}_{GraphSAGE}(v) = \theta_1 X_v + \theta_2 \cdot mean_{u \in N(v)}(X_u), \quad (15)$$

where θ_1 and θ_2 are two learnable parameters of GraphSAGE, $mean(\cdot)$ is an average function. In this work, we use an alternative option $sum(\cdot)$. Then, the loss function of node v which is labeled "1" is

$$\begin{aligned} \mathcal{L}(\theta; v, E) \approx & \big(\theta_1[2] X_v + \theta_2[2] \cdot \sum_{u \in N(v)} X_u \big) \\ & - \big(\theta_1[1] X_v + \theta_2[1] \cdot \sum_{u \in N(v)} X_u \big), \end{aligned} \quad (16)$$

where $\theta_1[c]$ and $\theta_2[c]$ correspond to the parameters for label "1" and label "2", respectively.

Similar to GCN, when we remove a set of edges E_{UL}, the loss of the node v by removing E_{UL} is formulated as

$$\mathcal{L}(\theta; v, E_{UL}) \approx \Big(\theta_2[2] \cdot \sum_{u \in \bar{N}(v)} X_u \Big) - \Big(\theta_2[1] \cdot \sum_{u \in \bar{N}(v)} X_u \Big), \quad (17)$$

where $\bar{N}(v) = \{u | (u, v) \in E_{UL}\}$.

And, the loss of the node v by the remaining edges $E \backslash E_{UL}$ is defined as:

$$\begin{aligned} \mathcal{L}(\theta; v, E \backslash E_{UL}) \approx & \big(\theta_1[2] X_v + \theta_2[2] \cdot \sum_{u \in N(v) \backslash \bar{N}(v)} X_u \big) \\ & - \big(\theta_1[1] X_v + \theta_2[1] \cdot \sum_{u \in N(v) \backslash \bar{N}(v)} X_u \big), \end{aligned} \quad (18)$$

Based on Eqns (16) - (18), we can have

$$\begin{aligned} \mathcal{L}(\theta; v, E \backslash E_{UL}) + \mathcal{L}(\theta; v, E_{UL}) = & (\theta_1[2] X_v - \theta_1[1]_X v) \\ & + (\theta_2[2] \cdot \sum_{u \in N(v) \backslash \bar{N}(v)} X_u + \theta_2[2] \cdot \sum_{u \in \bar{N}(v)} X_u) \\ & - (\theta_2[1] \cdot \sum_{u \in N(v) \backslash \bar{N}(v)} X_u + \theta_2[1] \cdot \sum_{u \in \bar{N}(v)} X_u). \end{aligned} \quad (19)$$

It is easy to see that

$$(N(v) \backslash \bar{N}(v)) \cup \bar{N}(v) = N(v). \quad (20)$$

Therefore, we prove that

$$\mathcal{L}(\theta; v, E \backslash E_{UL}) + \mathcal{L}(\theta; v, E_{UL}) = \mathcal{L}(\theta; v, E).$$

1.3 GIN

The aggregation method of GIN in node-wise can be formulated as

$$H_v = \mathsf{AGGREGATE}_{GIN}(v) = h_\theta \Big(\sum_{u \in N(v)} X_u + (1 + \varepsilon) X_v \Big), \quad (21)$$

where ε is a hyper-parameter of GIN, h_θ denotes a neural network, such as MLP. In this work, we only apply a linear as h_θ. Without losing generosity, we assume the true label of the given node v is label "1" (i.e., $y_{v,2} = 0$). Then we obtain the following loss function of node v by a GIN model,

$$\begin{aligned} \mathcal{L}(\theta; v, E) \approx & \Big(\sum_{u \in N(v)} X_u + (1 + \varepsilon) X_v \Big) \theta[2] \\ & - \Big(\sum_{u \in N(v)} X_u + (1 + \varepsilon) X_v \Big) \theta[1]. \end{aligned} \quad (22)$$

When we remove a set of edges E_{UL}, the loss of the node v by removing E_{UL} is formulated as

$$\mathcal{L}(\theta; v, E_{UL}) \approx \sum_{u \in \bar{N}(v)} X_u \theta[2] - \sum_{u \in \bar{N}(v)} X_u \theta[1], \quad (23)$$

where $\bar{N}(v) = \{u | (u, v) \in E_{UL}\}$.

Similarly, the loss of the node v by the remaining edges $E\backslash E_{UL}$ is defined as:

$$\begin{aligned}\mathcal{L}(\theta; v, E\backslash E_{UL}) \approx &(\sum_{u\in N(v)\backslash \bar{N}(v)} X_u + (1+\varepsilon)X_v)\theta[2] \\ &- (\sum_{u\in N(v)\backslash \bar{N}(v)} X_u + (1+\varepsilon)X_v)\theta[1].\end{aligned} \tag{24}$$

Based on Eqns (22) - (24), we can have:

$$\begin{aligned}\mathcal{L}(\theta; v, E\backslash E_{UL}) + \mathcal{L}(\theta; v, E_{UL}) = &(1+\varepsilon)X_v\theta[2] - (1+\varepsilon)X_v\theta[1] \\ &+ (\sum_{u\in N(v)\backslash \bar{N}(v)} X_u\theta[2] + \sum_{u\in \bar{N}(v)} X_u\theta[2]) \\ &- (\sum_{u\in N(v)\backslash \bar{N}(v)} X_u\theta[1] + \sum_{u\in \bar{N}(v)} X_u\theta[1])\end{aligned}$$

It is easy to see that

$$(N(v)\backslash \bar{N}(v)) \cup \bar{N}(v) = N(v). \tag{25}$$

Therefore, we prove that

$$\mathcal{L}(\theta; v, E\backslash E_{UL}) + \mathcal{L}(\theta; v, E_{UL}) = \mathcal{L}(\theta; v, E).$$