Table 8: Effectiveness results for LINE. Lower is better in gray columns. Higher is better in the others.

Datasets	Jaccard Index							Cosine Similarity						
	Diff	Orig. ROC	Fair ROC	Orig. F1	Fair F1	Reduce	Time	Diff	Orig. ROC	Fair ROC	Orig. F1	Fair F1	Reduce	Time
Twitch	1.079	0.687	0.691	0.625	0.622	1.92%	1878	1.267	0.687	0.662	0.625	0.606	12.1%	1999
PPI	0.674	0.682	0.678	0.618	0.620	2.06%	1656	0.699	0.682	0.686	0.618	0.621	1.22%	1779

Table 9: Effectiveness results for spectral clustering. Lower is better in gray columns. Higher is better in the others.

Datasets	Jaccard Index				Cosine Similarity			
	Diff	NMI	Reduce	Time	Diff	NMI	Reduce	Time
Twitch	0.031	1.000	5.44%	1698	0.107	1.000	24.5%	1714
PPI	1.035	0.914	19.5%	829.3	0.933	0.849	24.1%	985.1

mind, we can rewrite Eq. (5) as

$$\frac{\partial J}{\partial \tilde{\mathbf{A}}} = 2(\tilde{\mathbf{A}} - \mathbf{A}) + 2\alpha \left[\mathbf{r}'\mathbf{L_S}\frac{\partial \mathbf{r}}{\partial \tilde{\mathbf{A}}[i,j]}\right] \tag{20}$$

where $\mathbf{r} = (1-c)(\mathbf{I} - c\tilde{\mathbf{A}})^{-1}\mathbf{e}$. Based on [13], we have $\frac{\partial \mathbf{r}}{\partial \tilde{\mathbf{A}}[i,j]} = c\mathbf{r}[j](\mathbf{I} - c\tilde{\mathbf{A}})^{-1}[:,i]$. Then, define $\mathbf{Q} = (\mathbf{I} - c\tilde{\mathbf{A}})^{-1}$, we can further simplify Eq. (20) and get

$$\frac{\partial J}{\partial \tilde{\mathbf{A}}} = 2(\tilde{\mathbf{A}} - \mathbf{A}) + 2c\alpha\mathbf{Q}'\mathbf{L_S}\mathbf{r}\mathbf{r}' \tag{21}$$

Then, we can easily learn its debiased topology by applying Algorithm 1 with Eq. (21).

Algorithm 1 Instantiation with Spectral Clustering. For spectral clustering, as shown in Table 2, given an undirected graph with adjacency matrix $\mathbf{A}$, it finds the soft cluster membership matrix $\mathbf{U}$ as the eigenvectors of $\mathbf{L_A}$ associated with the smallest k eigenvalues. With that in mind, we first rewrite Eq. (5) as

$$\frac{\partial J}{\partial \tilde{\mathbf{A}}} = 2(\tilde{\mathbf{A}} - \mathbf{A}) + 2\alpha\frac{\partial \text{Tr}(\mathbf{U}'\mathbf{L_S}\mathbf{U})}{\partial \tilde{\mathbf{A}}} \tag{22}$$

However, directly calculating $\frac{\partial \text{Tr}(\mathbf{U}'\mathbf{L_S}\mathbf{U})}{\partial \tilde{\mathbf{A}}}$ is hard, we resort to chain rule. First, to calculate $\frac{\partial \text{Tr}(\mathbf{U}'\mathbf{L_S}\mathbf{U})}{\partial \mathbf{L}_{\tilde{\mathbf{A}}}}$, we denote $\mathbf{u}_i$ as the i^{th} column of $\mathbf{U}$ and write

$$\frac{\partial \text{Tr}(\mathbf{U}'\mathbf{L_S}\mathbf{U})}{\partial \mathbf{L}_{\tilde{\mathbf{A}}}} = 2\left[\text{Tr}((\mathbf{L_S}\mathbf{U})'\frac{\partial \mathbf{U}}{\partial \mathbf{L}_{\tilde{\mathbf{A}}}[i,j]})\right] = 2\sum_{i=1}^{k}\left[\mathbf{u}_i'\mathbf{L_S}\frac{\partial \mathbf{u}_i}{\partial \mathbf{L}_{\tilde{\mathbf{A}}}[i,j]}\right] \tag{23}$$

Denote $\mathbf{M}_i = (\lambda_i\mathbf{I} - \mathbf{L}_{\tilde{\mathbf{A}}})^+$ where λ_i is the i^{th} eigenvalue. Written in a matrix form, by the derivative of eigenvectors, we have

$$\left[\mathbf{u}_i'\mathbf{L_S}\frac{\partial \mathbf{u}_i}{\partial \mathbf{L}_{\tilde{\mathbf{A}}}[i,j]}\right] = \left[\mathbf{u}_i'\mathbf{L_S}\mathbf{M}[:,i]\mathbf{u}_i[j]\right] = \mathbf{M}_i'\mathbf{L_S}\mathbf{u}_i\mathbf{u}_i' \tag{24}$$

Then, based on [12], we get

$$\begin{aligned}\frac{\partial \text{Tr}(\mathbf{U}'\mathbf{L_S}\mathbf{U})}{\partial \tilde{\mathbf{A}}} &= diag\left(\frac{\partial \text{Tr}(\mathbf{U}'\mathbf{L_S}\mathbf{U})}{\partial \mathbf{L}_{\tilde{\mathbf{A}}}}\right)\mathbf{1}_{n\times n} - \frac{\partial \text{Tr}(\mathbf{U}'\mathbf{L_S}\mathbf{U})}{\partial \mathbf{L}_{\tilde{\mathbf{A}}}} \\ &= 2\sum_{i=1}^{k}\left(diag\left(\mathbf{M}_i'\mathbf{L_S}\mathbf{u}_i\mathbf{u}_i'\right)\mathbf{1}_{n\times n} - \mathbf{M}_i'\mathbf{L_S}\mathbf{u}_i\mathbf{u}_i'\right)\end{aligned} \tag{25}$$

where $\mathbf{1}_{n\times n}$ is an $n \times n$ matrix filled with 1. To learn the debiased topology, we can apply Algorithm 1 by combining Eq. (22) and (25).

D – Proof of Lemma 1

Proof. It takes $O(min\{m_1, m_2\})$ time to calculate $f(\mathbf{A} + \mathbf{A}') \circ \mathbf{L_S}$ and $O(m_2)$ time to calculate $diag(\mathbf{B}\mathbf{L_S})$. Thus the overall time complexity is $O(min\{m_1, m_2\} + m_2)$. For space complexity, it takes $O(min\{m_1, m_2\})$ space to save $f(\mathbf{A} + \mathbf{A}') \circ \mathbf{L_S}$ in sparse format and $O(n)$ space to save $diag(\mathbf{B}\mathbf{L_S})$. Therefore, the overall space complexity is $O(min\{m_1, m_2\} + n)$. □

E – Cost of Debiasing the Mining Model: A Case Study on PageRank

Given a graph with adjacency matrix $\mathbf{A}$, similarity matrix $\mathbf{S}$ and regularization parameter α, the cost of debiasing the mining model method on PageRank is summarized in Lemma 3.

Lemma 3. *Given a graph with the symmetric normalized adjacency matrix* $\mathbf{A}$ *and node-node similarity matrix* $\mathbf{S}$, *let* $\bar{\mathbf{r}}$ *be the PageRank vector without considering the fairness and* $\mathbf{r}^*$ *be the debiased PageRank vector as in Eq.* (12). *If teleportation vector* $\|\mathbf{e}\|_1 = 1$ *and similarity matrix* $\|\mathbf{S} - \mathbf{A}\|_F = \delta$, *it satisfies*

$$\|\mathbf{r}^* - \bar{\mathbf{r}}\|_F \leq \frac{2\alpha n}{1-c}(\delta + \sqrt{r(\mathbf{A})}\sigma_{max}(\mathbf{A}))$$

where c *is the damping factor and* α *is the regularization parameter for individual fairness.*

Proof. Recall that debiasing the mining model on PageRank is equivalent to solving the linear system $\mathbf{r} = c(\mathbf{A} - \frac{\alpha}{c}\mathbf{L_S})\mathbf{r} + (1-c)\mathbf{e}$. After rearranging terms, we can get its closed-form solution as $\mathbf{r}^* = (1-c)(\mathbf{I} - c\mathbf{A} + \alpha\mathbf{L_S})^{-1}\mathbf{e}$. If we do not consider individual fairness constraint, we can easily set $\mathbf{L_S} = \mathbf{0}$ and get $\bar{\mathbf{r}} = (1-c)(\mathbf{I} - c\mathbf{A})^{-1}\mathbf{e}$. Then we have the cost of individual fairness in PageRank as

$$\begin{aligned}\|\mathbf{r}^* - \bar{\mathbf{r}}\|_F &= (1-c)\|((\mathbf{I} - c\mathbf{A} + \alpha\mathbf{L_S})^{-1} - (\mathbf{I} - c\mathbf{A})^{-1})\mathbf{e}\|_F \\ &\leq (1-c)\|((\mathbf{I} - c\mathbf{A} + \alpha\mathbf{L_S})^{-1} - (\mathbf{I} - c\mathbf{A})^{-1})\|_F\|\mathbf{e}\|_F \\ &\leq (1-c)\|((\mathbf{I} - c\mathbf{A} + \alpha\mathbf{L_S})^{-1} - (\mathbf{I} - c\mathbf{A})^{-1})\|_F \\ &= (1-c)\|(\mathbf{I} - c\mathbf{A} + \alpha\mathbf{L_S})^{-1} \cdot \alpha\mathbf{L_S} \cdot (\mathbf{I} - c\mathbf{A})^{-1}\|_F \\ &\leq \alpha(1-c)\|(\mathbf{I} - c\mathbf{A} + \alpha\mathbf{L_S})^{-1}\|_F \cdot \|\mathbf{L_S}\|_F \cdot \|(\mathbf{I} - c\mathbf{A})^{-1}\|_F\end{aligned} \tag{26}$$

Since $\mathbf{A}$ is symmetric normalized matrix, its Laplacian matrix is $\mathbf{I} - \mathbf{A}$, which reveals that $\mathbf{I} - c\mathbf{A} = (1-c)\mathbf{I} + \mathbf{L}_{c\mathbf{A}}$ and $\mathbf{I} - c\mathbf{A} + \alpha\mathbf{L_S} = (1-c)\mathbf{I} + \mathbf{L}_{c\mathbf{A}+\alpha\mathbf{S}}$. Define $\mathbf{C} = (1-c)\mathbf{I} + \mathbf{L}_{c\mathbf{A}}$ and $\mathbf{D} = (1-c)\mathbf{I} + \mathbf{L}_{c\mathbf{A}+\alpha\mathbf{S}}$. Based on Eq. (18), we have the following two inequalities holds

$$\begin{aligned}\|(\mathbf{D})^{-1}\|_F &\leq \sqrt{n}\sigma_{\max}(((1-c)\mathbf{I} + \mathbf{L}_{c\mathbf{A}+\alpha\mathbf{S}})^{-1}) \\ &= \frac{\sqrt{n}}{\sigma_{\min}((1-c)\mathbf{I} + \mathbf{L}_{c\mathbf{A}+\alpha\mathbf{S}})} = \frac{\sqrt{n}}{1-c}\end{aligned} \tag{27}$$

$$\|(\mathbf{C})^{-1}\|_F \leq \sqrt{n}\sigma_{\max}(((1-c)\mathbf{I} + \mathbf{L}_{c\mathbf{A}})^{-1}) = \frac{\sqrt{n}}{1-c} \tag{28}$$

Combine Eq. (27), (28) with Eq. (26), we have $\|\mathbf{r}^* - \bar{\mathbf{r}}\|_F \leq \frac{\alpha n}{1-c}\|\mathbf{L_S}\|_F$. As shown in Eq. (19), we have $\|\mathbf{L_S}\|_F \leq 2(\delta + \sqrt{r(\mathbf{A})}\sigma_{\max}(\mathbf{A}))$. Thus, we have $\|\mathbf{r}^* - \bar{\mathbf{r}}\|_F \leq \frac{2\alpha n}{1-c}(\delta + \sqrt{r(\mathbf{A})}\sigma_{\max}(\mathbf{A}))$. □

Similar in Section 5, the cost of debiasing the mining model with PageRank depends on the number of nodes n, rank of adjacency matrix $r(\mathbf{A})$ and the largest singular value $\sigma_{\max}(\mathbf{A})$.