Adversarial Attacks on Node Embeddings via Graph Poisoning

Aleksandar Bojchevski, Stephan Günnemann

Overview
- Node embeddings are vulnerable to adversarial attacks.
- Exploit connections to matrix factorization and the graph spectrum to find adversarial edges.
- Relatively few perturbations degrade the embedding quality and the performance on downstream tasks.

Motivation
In domains where we use node embeddings (e.g., the Web) adversaries are common and false data is easy to inject. Research question: Are node embeddings robust to attacks?

Background: DeepWalk
Treat random walks as sentences. Train Word2Vec embeddings.

1. DeepWalk as Matrix Factorization
DeepWalk is equivalent to factorizing the Shifted Positive Pointwise Mutual Information (PPMI) matrix.

\[ A_u \lambda D_u S = U \Sigma^1/2 \]

Embeddings \( Z^* = U \Sigma_1^{1/2} \) obtained via SVD of \( M = U \Sigma V^T \)

2. Express the optimal \( \mathcal{L} \) via the graph spectrum
Rewrite \( S \) in terms of the generalized spectrum of \( A \). Optimal loss is a function of the eigenvalues \( \Rightarrow \) Inner optimization is eliminated.

\[ A u = \lambda D u \]

3. Approximate the poisoned graph’s spectrum
Compute the change using Eigenvalue Perturbation Theory.

\[ A_{\text{pois},} = A_{\text{clean}} + \Delta A \]

\[ \lambda_{\text{pois},} = \lambda_{\text{clean}} + u_{\text{clean}} (\Delta A + \lambda_{\text{clean}} \Delta D) u_{\text{clean}} \]

Overall algorithm:
1. Compute generalized eigenvalues/vectors \( (A / U) \) of the graph
2. For all candidate edge flips \( (i, j) \) compute the change in \( A / U \)
3. Greedily pick the top candidates leading to largest loss \( \mathcal{L} \)

General attack
Goal: decrease the overall quality of the embeddings.

Targeted attack
Goal: attack a specific node and/or a specific downstream task.

Transferability
Our selected adversarial edges transfer to other methods.

<table>
<thead>
<tr>
<th>budget</th>
<th>DW</th>
<th>SVD</th>
<th>SGNS</th>
<th>2vec</th>
<th>Spectral Embed.</th>
<th>Label Prop.</th>
<th>GCN</th>
</tr>
</thead>
<tbody>
<tr>
<td>250</td>
<td>-7.59</td>
<td>-5.73</td>
<td>-6.45</td>
<td>-3.58</td>
<td>-4.99</td>
<td>-2.21</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>-9.68</td>
<td>-11.47</td>
<td>-10.24</td>
<td>-4.57</td>
<td>-6.27</td>
<td>-8.61</td>
<td></td>
</tr>
</tbody>
</table>

Analysis of adversarial edges
There is no simple heuristic that can find the adversarial edges.