node2vec: Scalable Feature Learning for Networks

Arthur: Aditya Grover & Jure Leskovec

Presented by: Tao Ni
How to get good vector representations for nodes and edges?

- Classic unsupervised approaches:
  - Exploit spectral properties of various matrix representations of graphs (adjacency matrix, graph Laplacian)
    - PCA
    - IsoMap
  - Drawbacks:
    - Computationally expensive
    - Not robust to diverse patterns of graph.
Homophily

Nodes that are highly interconnected and belong to similar network communities should be embedded closely.
Structural Equivalence

Nodes that **have similar structural roles** in networks should be embedded closely together.
The notion of “Neighborhood”

- Homophily hypothesis emphasizes connectivity
- Structural Equivalence emphasizes structural roles
- In real-world, this two notions are not exclusive and graph commonly exhibit a mixture of both.
- E.g. Spectral Clustering makes a strong homophily assumption that graph cut will be useful for classification.
Node2vec: Feature learning framework

● An analogy of word2vec skip-gram model
  ○ But differs in the notion of “neighborhood”
Text vs Graph

The notion of neighborhood in a text is natural: context of the center word.

In graph, it’s a mixture of homophily and structural equivalence.
How to find a mixture of homophily neighbor and structurally equivalent neighbor for each node in the graph?
BFS Sampling

Neighborhood sampled by BFS leads to embeddings that correspond closely structural equivalence.

Figure 1: BFS and DFS search strategies from node $u$ ($k = 3$).
DFS Sampling

Reflects a macro-view of the neighborhood which is essential in inferring communities based on homophily.

Figure 1: BFS and DFS search strategies from node $u$ ($k = 3$).
Biased Random Walk

- Smooth interpolation between BFS and DFS
- See whiteboard
Return parameter, $p$

Controls likelihood of immediately revisiting a node in the walk.

If $p > \max(q, 1)$: moderate outward exploration.

If $p < \min(q, 1)$: keep the walk local, close to the starting node $u$.

Figure 2: Illustration of the random walk procedure in node2vec. The walk just transitioned from $t$ to $v$ and is now evaluating its next step out of node $v$. Edge labels indicate search biases $\alpha$. 
In-out parameter, $q$

Allows the search to differentiate between “inward” and “outward” nodes

If $q > 1$: biased to nodes close to node $t$, obtain a local view

If $q < 1$: encourage outward exploration

Figure 2: Illustration of the random walk procedure in node2vec. The walk just transitioned from $t$ to $v$ and is now evaluating its next step out of node $v$. Edge labels indicate search biases $\alpha$. 
Node2vec: Feature learning framework

- An analogy of word2vec skip-gram model
  - But differs in the notion of “neighborhood”
- See whiteboard.
Case study: Les Misérables network

$p = 1, q = 0.5$

homophily

$p = 1, q = 2$

structural equivalence
Experiment on multi-label classification for nodes (BlogCatalog)

- The performance improves as $p, q$ decreases
- Small $q$ encourages outward exploration while small $p$ ensures that the walk does not go too far from the start node
Experiment on multi-label classification for nodes (BlogCatalog)

- The performance improves as embedding dimension $d$ increases
- The performance saturated at $d$ equals around 128
Experiment on multi-label classification for nodes (BlogCatalog)

- The performance improves as the walk length, and avg walks per node increases.
Experiment on multi-label classification for nodes

- All these real-world graphs are a mix of both homophily and structural equivalence
- After adding flexibility in exploring neighborhoods allows node2vec to outperform the other benchmark algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>BlogCatalog</th>
<th>PPI</th>
<th>Wikipedia</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spectral Clustering</td>
<td>0.0405</td>
<td>0.0681</td>
<td>0.0395</td>
</tr>
<tr>
<td>DeepWalk</td>
<td>0.2110</td>
<td>0.1768</td>
<td>0.1274</td>
</tr>
<tr>
<td>LINE</td>
<td>0.0784</td>
<td>0.1447</td>
<td>0.1164</td>
</tr>
<tr>
<td>node2vec</td>
<td><strong>0.2581</strong></td>
<td><strong>0.1791</strong></td>
<td><strong>0.1552</strong></td>
</tr>
</tbody>
</table>

| node2vec settings (p,q) | 0.25, 0.25 | 4, 1 | 4, 0.5 |
| Gain of node2vec [%]   | **22.3** | **1.3** | **21.8** |

Table 2: Macro-F$_1$ scores for multilabel classification on BlogCatalog, PPI (Homo sapiens) and Wikipedia word cooccurrence networks with 50% of the nodes labeled for training.
Thank You!