Conditional Network Generation

Carl Yang, et al

jiyang3@illinois.edu
• Motivations
  • Massive networks decomposed into cells with clear structures and contexts
  • Map network structures and contextual semantics in embedding spaces
  • Flexibly generate network structures under given contexts

*Carl Yang, et al. NeurIPS’19* Conditional structure generation through graph variational generative adversarial nets. jiyang3@illinois.edu
• Leverage semantic relations among contexts in the CubeNet
• Learn from data-rich CubeNet cells and generate for data-scarce ones
• Generalizable to social networks, sensor networks, and so on
• Challenge 1: Flexible context-structure conditioning
  • Learn a single representation from a set of graphs with variable sizes
Conditional Network Generation

**Challenge 2:** Permutation-invariant graph generation

- Capture unique graph representations regardless of node ordering

\[
\begin{array}{c}
| & 1 & 2 & 3 & 4 \\
1 & 2 & 3 & 4 \\
2 & 3 & 4 & 1 \\
3 & 4 & 1 & 2 \\
4 & 1 & 2 & 3 \\
0 & 0 & 1 & 1 \\
0 & 0 & 1 & 0 \\
1 & 1 & 0 & 1 \\
1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 \\
1 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 \\
1 & 1 & 1 & 0 \\
\end{array}
\]

\[
\begin{array}{c}
A_1 & A_2 & A_3 \\
\end{array}
\]

\[
\begin{array}{c}
||A_1 - A_2||^2 = 2.8284 \\
||A_1 - A_3||^2 = 2.2361 \\
||A_1 - A_2||^2 = 1.7321 \\
\end{array}
\]
Conditional Network Generation

• **Technical contribution 1**
  
  • A novel GCN-VAE framework with flexible conditioning function
    
    • Use a single distribution to jointly model all nodes
    
    $$
    q(z_i|X, A) \sim \mathcal{N}(\bar{z}|\bar{\mu}, \text{diag}(\bar{\sigma}^2)), \text{ where } \bar{\mu} = \frac{1}{n} \sum_{i=1}^{n} g_{\mu}(X, A)_i, \bar{\sigma}^2 = \frac{1}{n^2} \sum_{i=1}^{n} g_{\sigma}(X, A)_i^2.
    $$

  Learn with sets of graphs
  Graph-level conditioning
  Permutation-invariant encoding
**Technical contributions**

- GCN-based graph discriminator to enable permutation invariance
  - End-to-end learnable graph structure representations

\[ \mathcal{L}_{gan} = \log(D(A)) + \log(1 - D(A')) , \text{ with } D(A) = f'(g'(X(A), A)) \]

\[ \begin{align*} U_1 &\neq U_2 \\
U_2 &\equiv U_3 \end{align*} \]
Conditional Network Generation

• Overall Framework

\[ \mathcal{L}_{gvgan} = \mathcal{L}_{rec} + \lambda_1 \mathcal{L}_{prior} + \lambda_2 \mathcal{L}_{gan} \]

\[ \theta_E \leftarrow -\nabla_{\theta_E} (\mathcal{L}_{rec} + \lambda_1 \mathcal{L}_{prior}), \quad \theta_G \leftarrow -\nabla_{\theta_G} (\mathcal{L}_{rec} - \lambda_2 \mathcal{L}_{gan}), \quad \theta_D \leftarrow -\nabla_{\theta_D} \lambda_2 \mathcal{L}_{gan} \]
**Theoretical discussions**

- Representative and permutation-invariant graph encoding and discrimination
  \[
  \alpha d(G, G') \leq d(\mathcal{E}(A), \mathcal{E}(A')) \leq \beta d(G, G')
  \]

- Permutation-invariant graph generation
  
  **Encoding:** \(\mathcal{E}(PAP^T) = \mathcal{E}(A)\);  
  **Decoding:** \(\mathcal{L}(A', A) = \mathcal{L}(PA'P^T, A)\).

- Learnable reconstruction loss and mapping consistency
  
  \(\mathcal{E}(\mathcal{G}(\mathcal{E}(A')) \rightarrow \mathcal{E}(\mathcal{G}(\xi)) \rightarrow \mathcal{E}(A), \quad \xi \in Z \odot C\)
### Conditional Network Generation

#### Experimental results

<table>
<thead>
<tr>
<th>Models</th>
<th>LCC</th>
<th>TC</th>
<th>CPL</th>
<th>MD</th>
<th>GINI</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Real</strong></td>
<td>96.00</td>
<td>48.54</td>
<td>3.696</td>
<td>11.62</td>
<td>0.3293</td>
</tr>
<tr>
<td><strong>DBLP Seen</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GVAE</td>
<td>20.91**</td>
<td>21.76**</td>
<td>1.390*</td>
<td>2.32**</td>
<td>0.1964**</td>
</tr>
<tr>
<td>NetGAN</td>
<td>21.15**</td>
<td>22.46**</td>
<td>1.641**</td>
<td>2.77**</td>
<td>0.0568**</td>
</tr>
<tr>
<td>GraphRNN</td>
<td>6.88*</td>
<td>69.32**</td>
<td>1.628**</td>
<td>7.06**</td>
<td>0.2446**</td>
</tr>
<tr>
<td>CONGEN(R)</td>
<td>6.70*</td>
<td>7.70*</td>
<td>1.201*</td>
<td>1.33</td>
<td>0.1232*</td>
</tr>
<tr>
<td>CONGEN(S)</td>
<td><strong>6.00</strong></td>
<td>11.32</td>
<td><strong>0.963</strong></td>
<td>1.48</td>
<td>0.0959</td>
</tr>
<tr>
<td><strong>DBLP Unseen</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real</td>
<td>102.50</td>
<td>58.21</td>
<td>4.982</td>
<td>14.29</td>
<td>0.3223</td>
</tr>
<tr>
<td>GVAE</td>
<td>17.40**</td>
<td>17.02**</td>
<td>1.521**</td>
<td>3.53*</td>
<td>0.2479**</td>
</tr>
<tr>
<td>NetGAN</td>
<td>29.57**</td>
<td>39.85**</td>
<td>1.494**</td>
<td>3.71**</td>
<td><strong>0.0812</strong></td>
</tr>
<tr>
<td>GraphRNN</td>
<td>6.43</td>
<td>73.21**</td>
<td>1.305*</td>
<td>6.43**</td>
<td>0.1447**</td>
</tr>
<tr>
<td>CONGEN(R)</td>
<td>9.25*</td>
<td>10.50</td>
<td>1.445**</td>
<td><strong>1.92</strong></td>
<td>0.1418**</td>
</tr>
<tr>
<td>CONGEN(S)</td>
<td><strong>6.33</strong></td>
<td><strong>10.17</strong></td>
<td><strong>1.162</strong></td>
<td><strong>1.92</strong></td>
<td>0.0861</td>
</tr>
<tr>
<td><strong>TCGA Seen</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real</td>
<td>177.34</td>
<td>8913.20</td>
<td>4.171</td>
<td>38.27</td>
<td>0.4192</td>
</tr>
<tr>
<td>GVAE</td>
<td>54.82**</td>
<td>2396.94*</td>
<td>1.538</td>
<td>14.10**</td>
<td>0.2035**</td>
</tr>
<tr>
<td>NetGAN</td>
<td>32.02**</td>
<td>3614.61**</td>
<td>1.702**</td>
<td>17.61**</td>
<td>0.1289*</td>
</tr>
<tr>
<td>GraphRNN</td>
<td><strong>16.20</strong></td>
<td>2881.68**</td>
<td>1.899**</td>
<td>18.78**</td>
<td>0.2726**</td>
</tr>
<tr>
<td>CONGEN(R)</td>
<td>34.42**</td>
<td>2594.16**</td>
<td>1.542</td>
<td>9.50</td>
<td>0.1590**</td>
</tr>
<tr>
<td>CONGEN(S)</td>
<td>23.72</td>
<td><strong>2076.05</strong></td>
<td><strong>1.524</strong></td>
<td><strong>8.32</strong></td>
<td><strong>0.1093</strong></td>
</tr>
<tr>
<td><strong>TCGA Unseen</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Real</td>
<td>177.91</td>
<td>8053.18</td>
<td>4.143</td>
<td>34.34</td>
<td>0.4154</td>
</tr>
<tr>
<td>GVAE</td>
<td>37.18**</td>
<td>2768.55**</td>
<td><strong>1.324</strong></td>
<td>13.03**</td>
<td>0.1497**</td>
</tr>
<tr>
<td>NetGAN</td>
<td>31.36**</td>
<td>3557.91**</td>
<td>1.645*</td>
<td>18.45**</td>
<td>0.1277**</td>
</tr>
<tr>
<td>GraphRNN</td>
<td><strong>15.73</strong></td>
<td>2605.73**</td>
<td>1.859**</td>
<td>13.55**</td>
<td>0.2647**</td>
</tr>
<tr>
<td>CONGEN(R)</td>
<td>27.77*</td>
<td>3083.81**</td>
<td>1.362*</td>
<td>10.86*</td>
<td>0.1413**</td>
</tr>
<tr>
<td>CONGEN(S)</td>
<td>23.97</td>
<td><strong>2058.95</strong></td>
<td>1.522</td>
<td><strong>8.68</strong></td>
<td><strong>0.1003</strong></td>
</tr>
</tbody>
</table>
Conditional Network Generation

Condition vector: [0, 0, 0, 0, 1, 0, 0, 2, 3]
Semantics: CIRK, high productivity, 2000-2009
LCC: 76, TC: 5, CPL: 6.115, MD: 10, GIN: 0.3329
Seen during training

(a) Real graph
(b) Generated graphs by GVAE
(c) Generated graphs by NetGAN
(d) Generated graphs by GraphRNN
(e) Generated graphs by GVGAN

Condition vector: [0, 0, 0, 0, 0, 0, 0, 2, 2]
Semantics: ICDM, mid productivity, 2000-2009
LCC: 20, TC: 3, CPL: 2.9, MD: 10, GIN: 0.2665
Seen during training

(a) Real graph
(b) Generated graphs by GVAE
(c) Generated graphs by NetGAN
(d) Generated graphs by GraphRNN
(e) Generated graphs by GVGAN

Condition vector: [0, 0, 0, 1, 0, 0, 0, 0, 3, 1]
Semantics: SIGIR, low productivity, 2010-2019
LCC: 12, TC: 1, CPL: 2.379, MD: 6, GIN: 0.1486
Unseen during training

(a) Real graph
(b) Generated graphs by GVAE
(c) Generated graphs by NetGAN
(d) Generated graphs by GraphRNN
(e) Generated graphs by GVGAN

jiyang3@illinois.edu
• For more details, please refer to the following links
• Full paper: http://jiyang3.web.engr.illinois.edu/files/condgen.pdf
• Code and data: https://github.com/KelestZ/CondGen
• Supplementary materials: http://jiyang3.web.engr.illinois.edu/files/condgen_sup.pdf
• Poster: http://jiyang3.web.engr.illinois.edu/files/condgen_poster.pdf

Thanks!