The Infinity Mirror Test for Graph Generators

Satyaki Sikdar · Daniel Gonzalez · Tim Weninger

Department of Computer Science & Engineering
University of Notre Dame
{ssikdar, dgonza26, tweninge}@nd.edu

Introduction

- Graph generative models are a way of learning the salient features of a given graph to generate similar graphs that capture the original's essence.
- However, these models make assumptions during feature extraction and generation that may not be apparent in the generated graph.
- By repeatedly fitting the same model to the graphs it generates, the model's implicit biases will be amplified and exaggerated.

Framework

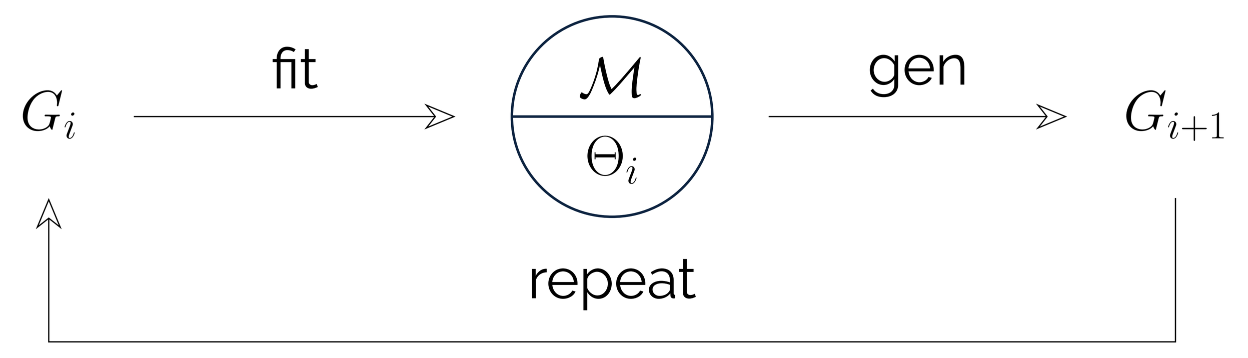
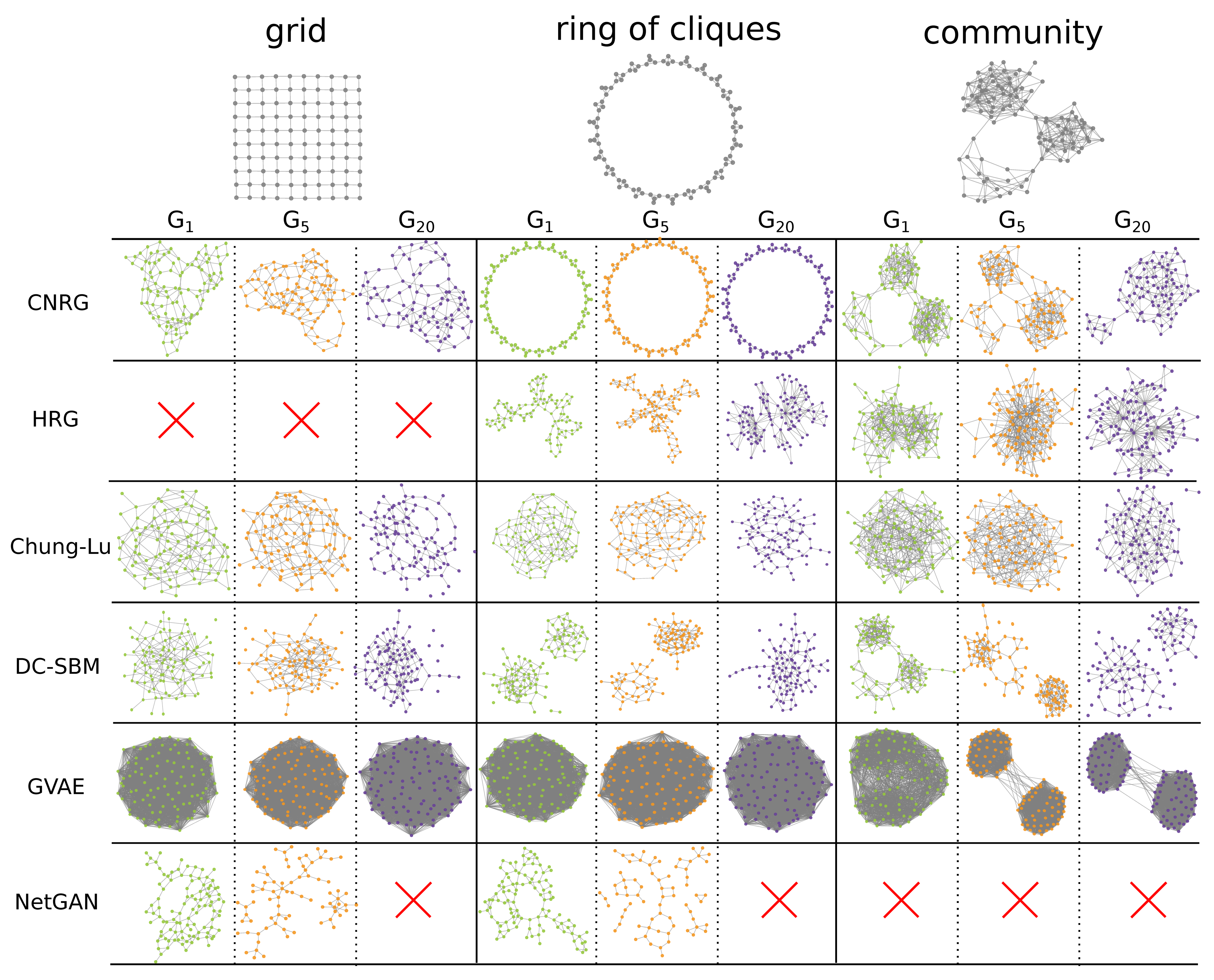


Figure 1. Fit a model $\mathcal{M}$ on an input graph G_i to learn Θ_i and generate G_{i+1}. Repeat the process on G_{i+1}.

Methodology

- Given a choice of model $\mathcal{M}$ and an initial seed graph G_0, we can use the idea in Figure 1 to generate a sequence of n graphs $G_1, G_2, \cdots, G_n$.
- The degeneration experienced by a chain can be quantified by comparing the initial graph G_0 to the last graph G_n using a graph similarity metric such as DeltaCon.
- For each $(\mathcal{M}, G_0)$ pair, we generate 50 chains, select the one with median DeltaCon score, and visualize G_1, G_5, and G_{20} (i.e., the 1st, 5th, and 20th generations) in the figure to the right.



Key Findings

- CNRG does well on graphs with community structure but not on highly-regular graphs.
- HRG's grammar extraction fails on grids and fares poorly on the other two graph types.
- Chung-Lu entirely fails to capture local and global network structure.
- SBM performs similarly to CNRG on grids and community but worse on the clique ring.
- GraphVAE produces overly dense graphs regardless of the input.
- NetGAN tends to generate increasingly sparse graphs until failure.

The authors are supported by a grant from the US National Science Foundation (#1652492).

WENINGER LAB

UNIVERSITY OF NOTRE DAME