GRAPH GENERATION WITH SPECIFIED PROPERTIES

Rosalie Millner, Emilio Picard
Master MVA, Department of Mathematics, ENS Paris Saclay, exhibited at MBZUAI France Lab Generative AI workshop in Paris, February 12-13 2025

école normale supérieure paris—saclay

Dauphine | PSL

SORBONNE UNIVERSITÉ

MOTIVATION

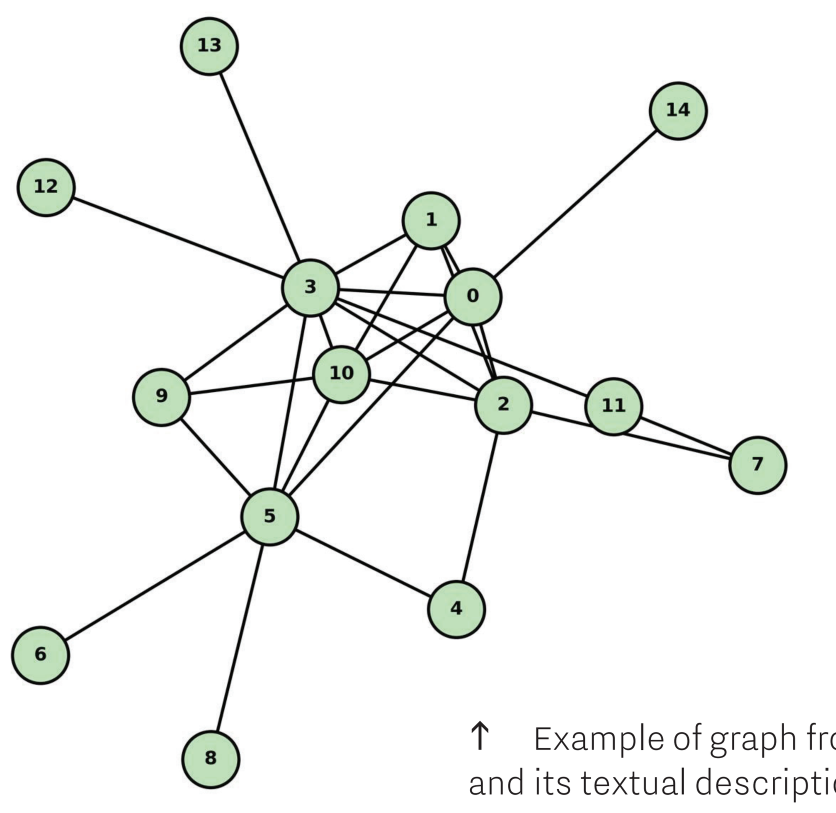
- Increasing interest in generating graphs given specified properties
- Applications from bio-chemistry to social networks and infrastructure modelling
- Complexity and high-dimensional nature of graph properties make graph generation challenging

We explore conditional graph generation using latent diffusion models*. By conditioning the generation process on graph properties extracted from textual descriptions, the goal is to produce graphs that accurately match with the specified characteristics.

* Based on the paper *"Neural graph generator: Feature-conditioned graph generation using latent diffusion models"* by Evdaimon, Nikolentzos et al., 2024 [1].

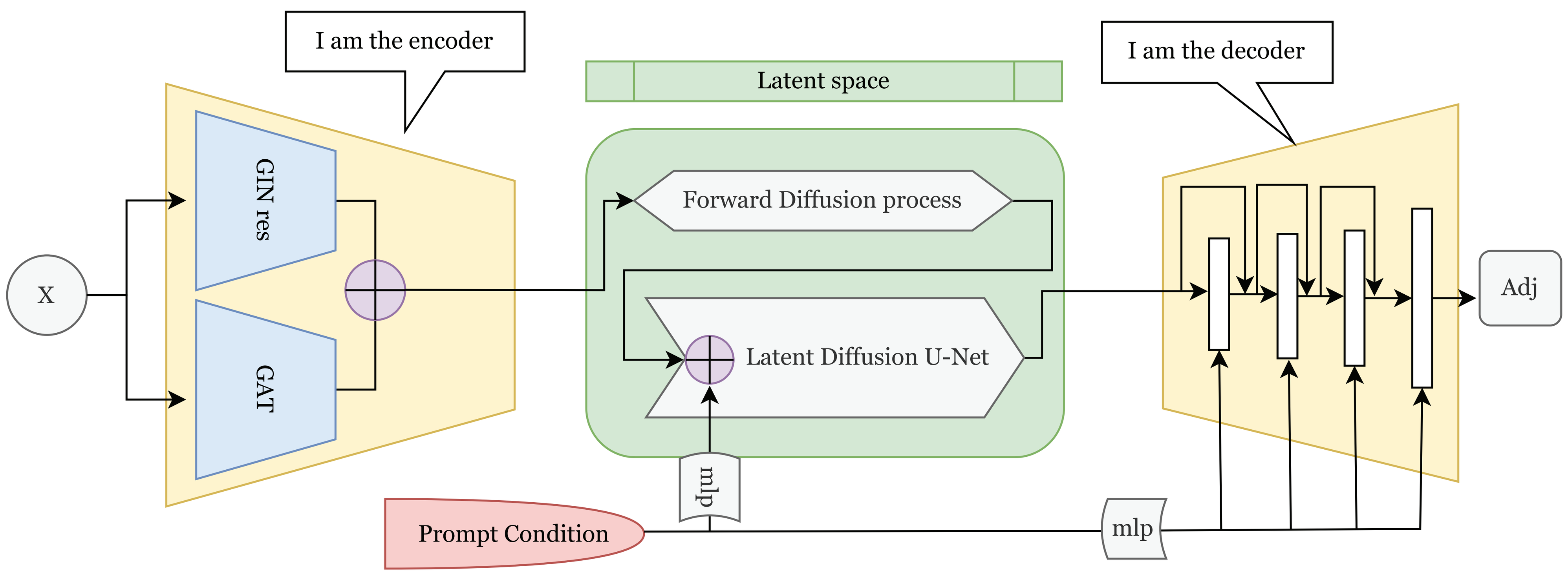
DATA PREPROCESSING

Input textual description:
"This graph comprises 15 nodes and 26 edges. The average degree is equal to 3.467 and there are 16 triangles in the graph. The global clustering coefficient and the graph's maximum k-core are 0.421 and 4 respectively. The graph consists of 3 communities."



↑ Example of graph from the training set and its textual description

Baseline Features	Value
Number of nodes	15
Number of edges	26
Average node degree	3.467
Number of triangles	16
Global clustering coeff	0.421
Maximum k-core	4
Number of communities	3
Our additional features	
Value Density	0.248
Triangle Formation Ratio	0.035
Edge-to-Node	1.733



METHOD

2 steps training process:

- **Autoencoder training** — Loss: ℓ_1 (reduction=sum), during 600 epochs.
- **Denoiser training** — Loss: Smooth ℓ_1, during 600 epochs.

Use of AdamW Optimizer for each training step.

EVALUATION

Metric used for evaluation:

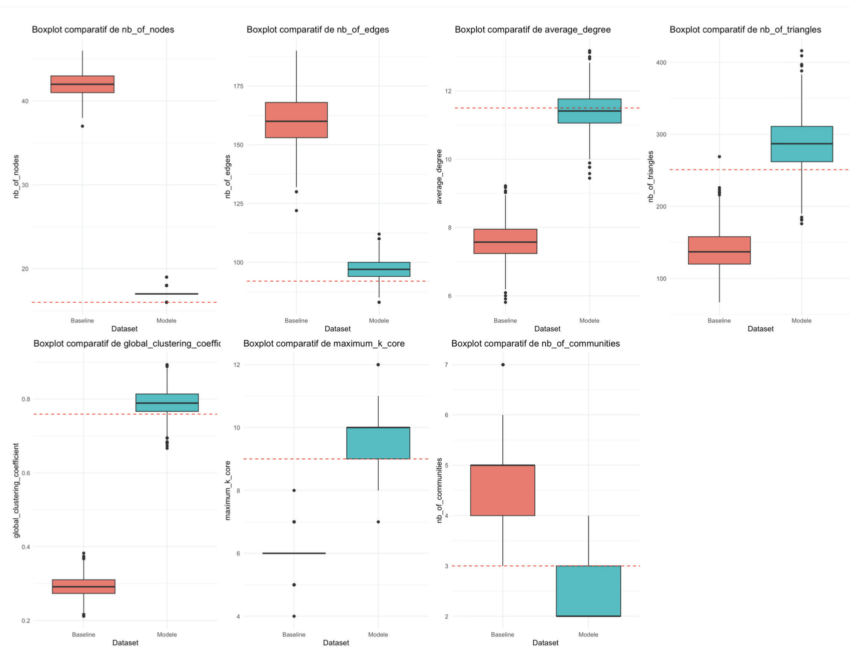
$\mathrm{MAE}(\hat{G}) = \frac{1}{N}\sum_{i=1}^{N}(\hat{G}^i - G^i_{true})$,

where $\hat{G}$ and G_{true} are the extracted vectors of normalized properties for the generated graph and the true input condition, respectively.

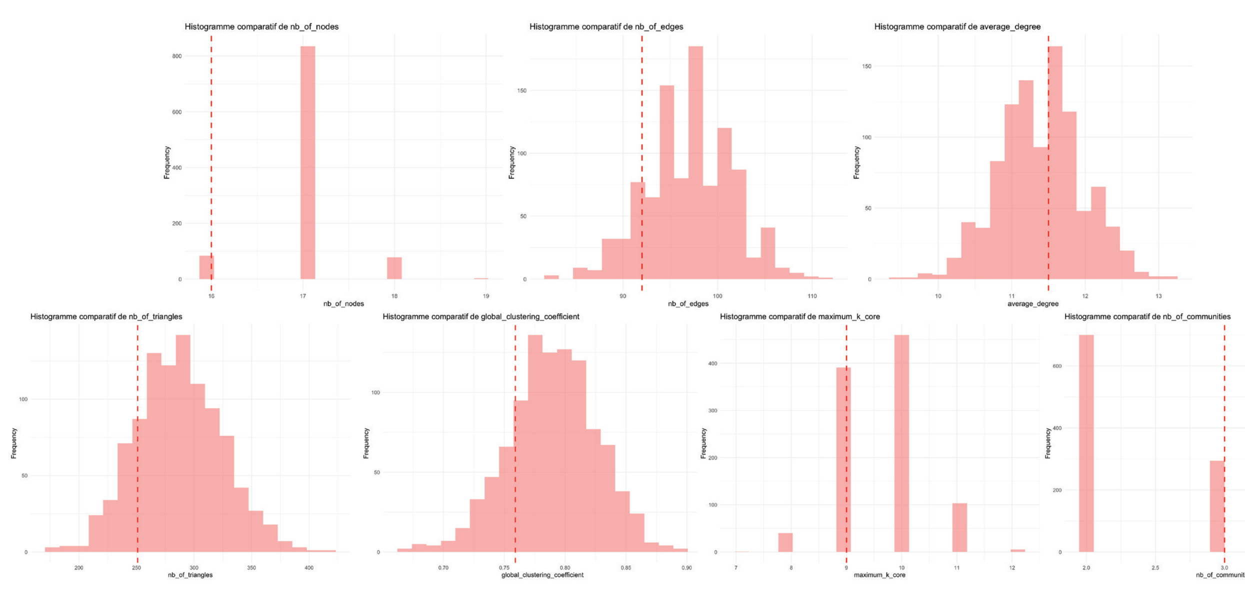
Method	MAE
GIN-GAT + Normal Decoder (n_cond=7)	0.0061
GIN-GAT + Residual Decoder (n_cond=7)	0.0058
GIN-GAT + Residual Decoder (n_cond=10)	**0.0056**

Given the observed variance within our generations, we opted for a graph selection strategy, inspired from acceptation-rejection Bayesian approaches.

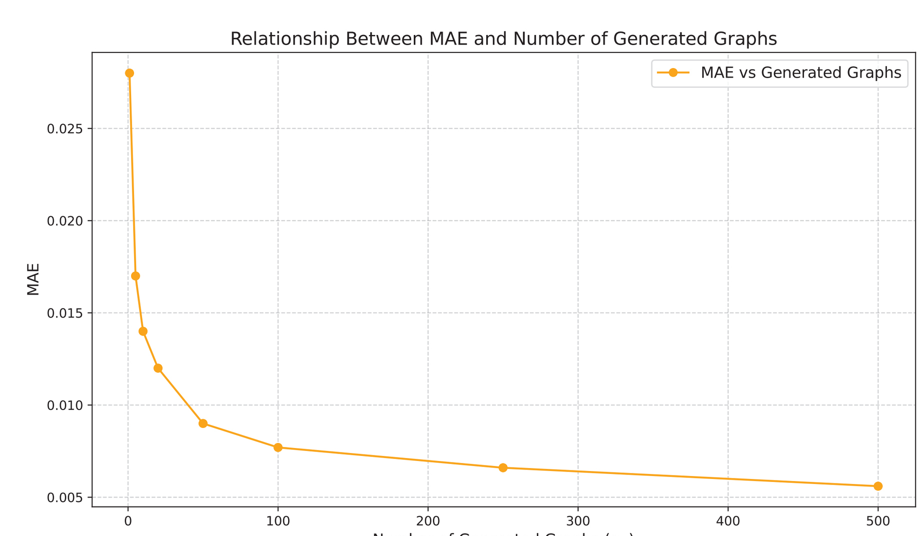
Result statistics



↑ Boxplots of the features of 1000 generations for a graph (red: baseline model, blue: our model, dotted red line: ground truth)



↑ Histograms of the features for 1000 graph generations with our model



There is an important reduction in MAE as the number of generated graphs increases. For computational efficiency, we set n_G = 100 generated graphs for selection.

← Analysis of the impact of varying the number of generated graphs on model performance

LIMITATIONS The main limitation of this model is that it relies on very strong assumptions regarding the input textual conditions. Specifically, if the values of the properties are given in a different order, it will significantly affect the quality of the generated outputs. However, this issue lies outside the scope of our current problem, and for the task at hand, we have achieved encouraging results.

REFERENCES Iakovos Evdaimon *et al.* *"Neural Graph Generator: Feature-Conditioned Graph Generation using Latent Diffusion Models"*, 2024.

The code and project report are available at **github.com/emilio-pcrd/graph-generation**