Research notes on metrics for GNNs applied to biological problems

Philip Hartout

Monday January 24, 2022

1 Graph Neural Networks

1.1 Reviews

1.1.1 Graph neural networks:

A review of methods and applications Zhou et al mention several GNN approaches in use today Generative models popular today:

Sequential graph generation process

- GraphRNN - generates the adjacency matrix of a graph by generating the adjacency vector of each node step by step, with graph outputs with different number of nodes.
- Li 2018 - also generates nodes and edges sequentially uses the hidden state to decide what to do at the next step
- GraphAF - also a sequential process, Conducts a validity check of each molecule generated at each step to see if it's valid.

Non-sequential graph generation process

- MolGAN - to generate small molecules. Uses a permutation-invariant to solve the node adjacency matrix at once. Also implements an RL-based optimization toward desired chemical properties
- Ma et al 2018 - constrained VAE for semantic validity of generated graph
- GCPN similar to MolGAN, uses RL based methods to ensure validity of domain-specific rules
- Graph Normalizing Flows

This one has a fairly comprehensive website: `https://sites.google.com/view/graph-normalizing-f`
Full architecture

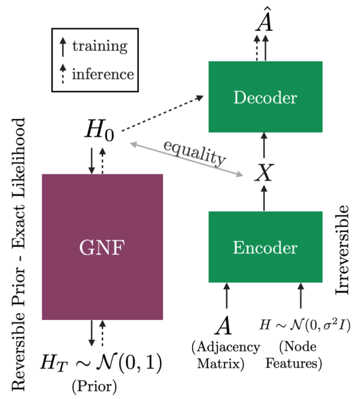

Figure 1: figure name

- Graphite isotropic gaussian for VAE + iterative refinement for decoding

1.2 Three most popular according to O'Bray 2021:

- GraphRNN, GRAN, Graph Score Matching.
- Graph Recurrent Attention Networks

> In previous work, You et al. [37] computed degree distributions, clustering coefficient distributions, and the number of occurrence of all orbits with 4 nodes, and then used the maximum mean discrepancy (MMD) over these graph statistics, relying on Gaussian kernels with the first Wasserstein distance, i.e., earth mover's distance (EMD), in the MMD.In practice, we found computing this MMD with the Gaussian EMD kernel to be very slow for moderately large graphs. Therefore, we use the total variation (TV) distance, which greatly speeds up the evaluation and is still consistent with EMD. In addition to the node degree, clustering coefficient and orbit counts (used by [36]), we also compare the spectra of the graphs by computing the eigenvalues of the normalized graph Laplacian (quantized to approximate a probability density). This spectral comparison provides a view of the global graph properties, whereas the previous metrics focus on local graph statistics.

- Graph Score Matching On MMD, they say the following:

2 Generative modelling metrics

a common practice of applying Langevin dynamics. We chose the value of the hyper-parameters based on the MMD metrics on the validation set, which contains 32 samples from the training set.

$$\tilde{\mathbf{x}}_t \leftarrow \tilde{\mathbf{x}}_{t-1} + \frac{\alpha_i}{2}\mathbf{s}_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}_{t-1}, \sigma_i) + \epsilon_s\sqrt{\alpha_i}\,\mathbf{z}_t$$

Figure 2: MMD optimization strategy

2.1 Objective:

2.1.1 Generative graph dist close to the input graph dist

2.1.2 (pseudo)-metric to assess dissimilarity between G (generated graphs) and G* (input graphs)

2.2 On images

2.2.1 Frechet Inception Distance

The idea here is to use deeper representational layers of an ANN and used the squared Wasserstein metric to compare two multinomial Gaussians. Introduced 2017

2.2.2 LPIPS Project page

Introduced 2017

2.2.3 Why comparing graphs is hard:

- Metrics need to deal with spatial invariances such as cycles.
- Graph edit distance is NP-hard (Zeng 2009) and therefore does not satisfy efficiency criterion.
- Other publications:

2.3 Desiderata for good metrics:

1. Robust to noise
2. Expressive, if they don't arise from the same dist, then metric should detect this.
3. Computationally efficient.

3 MMD - current accepted method to evaluate generative GNNs

- The MMD formula goes as follows:

$\mathrm{MMD}(X,Y) := \frac{1}{n^2}\sum_{i,j=1}^{n} k(x_i,x_j) + \frac{1}{m^2}\sum_{i,j=1}^{n} k(y_i,y_j) - \frac{2}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} k(y_i,y_j)$

- use it for hypothesis/two-sample testing.
- In practice, we evaluate $d_{MMD}(\mathcal{G},\mathcal{G}*) := MMD(f(\mathcal{G}), f(\mathcal{G}*))$ for a distribution $\mathcal{G}$. Given multiple distributions $G_1, G_2, \ldots$, the values of d_{MMD} can be used to rank models, where smaller values are assumed to indicate a larger agreement with the original distribution $\mathcal{G}*$.
- Commonly used kernels: first Wasserstein distance, total variation distance, radial basis function.
- Commonly used descriptor functions: degree distribution histogram, clustering coefficient, Laplacian spectrum histogram.

3.1 Potential pitfalls of descriptors

- Degree distributions are ok seemingly
- Clustering does not distinguish fully connected vs disconnected cliques
- Spectral methods are not clearly expressive. Does not seem to be for certain classes of graphs.
- Parameters and descriptors are set a priori in the best case
- Model performance is highly dependent on parameters and descriptor functions.

4 Research objectives

There are multiple objectives here:

1. Find optimal kernel/hyperparameter combination based on controlled experiments on a given dataset to evaluate a good MMD configuration.
    - For this we will need `https://www.alphafold.ebi.ac.uk/download`, because it's clean. Also filter single chain proteins to extract graphs in the first place.
    - This can be built as a first step to get the pipeline going.
2. Show which parameters influence evaluation and how?
    - Conduct perturbation experiments on graphs
3. Find novel domain-agnostic evaluation & domain-specific evaluation metrics
    (a) Domain-agnostic evaluation measures
        - Correlation with graph-edit distance

- Correlation with perturbation
- Topology/persistence based approaches could be useful for modelling features like binding pockets, etc?

(b) Domain-specific evaluation measures

- Alignment
- Energy?

References

References

[1] Nicola De Cao and Thomas Kipf. "MolGAN: An implicit generative model for small molecular graphs". In: *arXiv preprint arXiv:1805.11973* (2018).

[2] Aditya Grover, Aaron Zweig, and Stefano Ermon. "Graphite: Iterative generative modeling of graphs". In: *International conference on machine learning*. PMLR. 2019, pp. 2434–2444.

[3] Martin Heusel et al. "Gans trained by a two time-scale update rule converge to a local nash equilibrium". In: *Advances in neural information processing systems* 30 (2017).

[4] Yujia Li et al. "Learning deep generative models of graphs". In: *arXiv preprint arXiv:1803.03324* (2018).

[5] Renjie Liao et al. "Efficient graph generation with graph recurrent attention networks". In: *arXiv preprint arXiv:1910.00760* (2019).

[6] Jenny Liu et al. "Graph normalizing flows". In: *arXiv preprint arXiv:1905.13177* (2019).

[7] Tengfei Ma, Jie Chen, and Cao Xiao. "Constrained generation of semantically valid graphs via regularizing variational autoencoders". In: *arXiv preprint arXiv:1809.02630* (2018).

[8] Chenhao Niu et al. "Permutation invariant graph generation via score-based generative modeling". In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pp. 4474–4484.

[9] Chence Shi et al. "GraphAF: a flow-based autoregressive model for molecular graph generation". In: *arXiv preprint arXiv:2001.09382* (2020).

[10] Lucas Theis, Aäron van den Oord, and Matthias Bethge. "A note on the evaluation of generative models". In: *arXiv preprint arXiv:1511.01844* (2015).

[11] Jiaxuan You et al. "Graph convolutional policy network for goal-directed molecular graph generation". In: *arXiv preprint arXiv:1806.02473* (2018).

[12] Jiaxuan You et al. "GraphRNN: Generating realistic graphs with deep auto-regressive models". In: *International conference on machine learning*. PMLR. 2018, pp. 5708–5717.

[13] Richard Zhang et al. "The unreasonable effectiveness of deep features as a perceptual metric". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 586–595.

[14] Jie Zhou et al. "Graph neural networks: A review of methods and applications". In: *AI Open* 1 (2020), pp. 57–81.