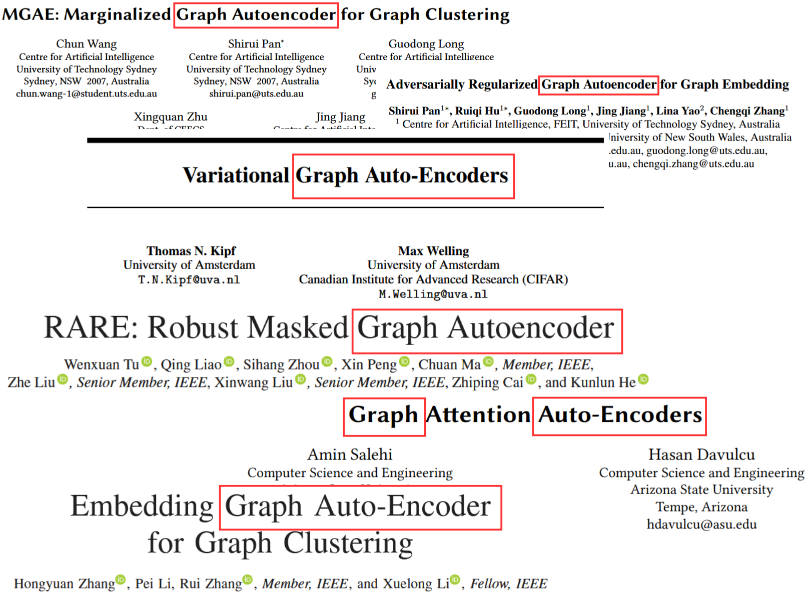
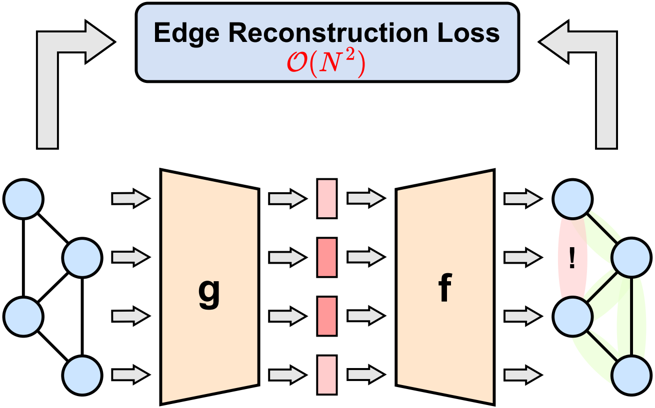
The quest for the GRAph Level autoEncoder.

NeurIPS 2025

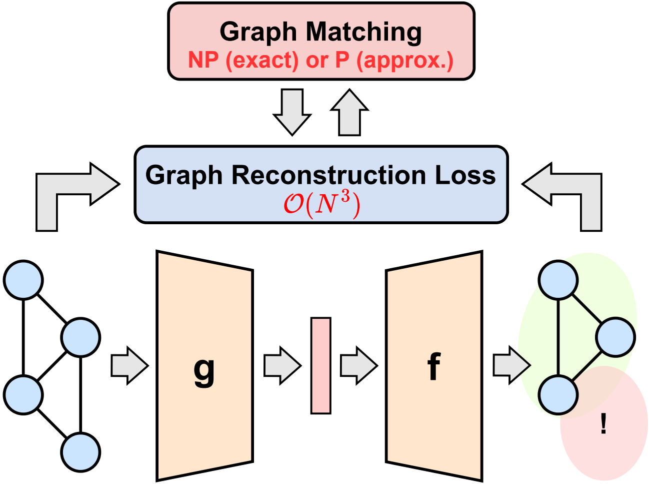
Paul Krzakala, Gabriel Melo,
Charlotte Laclau, Florence d'Alché-Buc, Rémi Flamary

Ecole Polytechnique (CMAP) & Télécom Paris (LTCI).

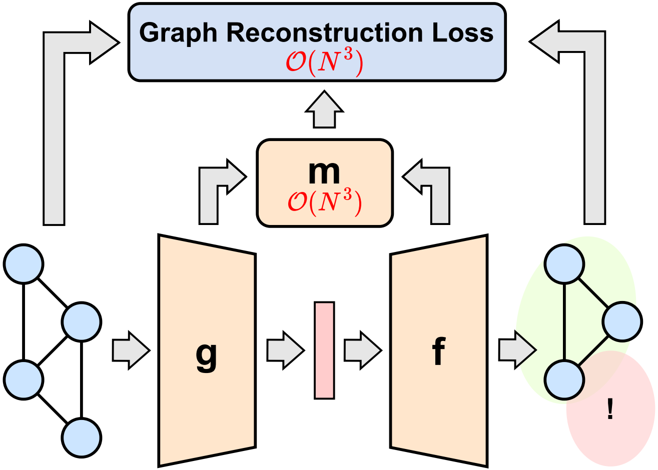

MGAE: Marginalized Graph Autoencoder for Graph Clustering
Chun Wang
Centre for Artificial Intelligence
University of Technology Sydney
Sydney, NSW 2007, Australia
chun.wang-1@student.uts.edu.au
Shirui Pan*
Centre for Artificial Intelligence
University of Technology Sydney
Sydney, NSW 2007, Australia
shirui.pan@uts.edu.au
Guodong Long
Centre for Artificial Intelligence
Xingquan Zhu
Jing Jiang
Adversarially Regularized Graph Autoencoder for Graph Embedding
Shirui Pan1*, Ruiqi Hu1*, Guodong Long1, Jing Jiang1, Lina Yao2, Chengqi Zhang1
1 Centre for Artificial Intelligence, FEIT, University of Technology Sydney, Australia
.edu.au, guodong.long@uts.edu.au,
u.au, chengqi.zhang@uts.edu.au
Variational Graph Auto-Encoders
Thomas N. Kipf
University of Amsterdam
T.N.Kipf@uva.nl
Max Welling
University of Amsterdam
Canadian Institute for Advanced Research (CIFAR)
M.Welling@uva.nl
RARE: Robust Masked Graph Autoencoder
Wenxuan Tu, Qing Liao, Sihang Zhou, Xin Peng, Chuan Ma, Member, IEEE,
Zhe Liu, Senior Member, IEEE, Xinwang Liu, Senior Member, IEEE, Zhiping Cai, and Kunlun He
Graph Attention Auto-Encoders
Amin Salehi
Computer Science and Engineering
Hasan Davulcu
Computer Science and Engineering
Arizona State University
Tempe, Arizona
hdavulcu@asu.edu
Embedding Graph Auto-Encoder
for Graph Clustering
Hongyuan Zhang, Pei Li, Rui Zhang, Member, IEEE, and Xuelong Li, Fellow, IEEE


Edge Reconstruction Loss
$\mathcal{O}(N^2)$
g
f
!

Graph-Level Autoencoders (Naive)



Graph-Level Autoencoders (GRALE)



Is it reasonable?

$\rightarrow$ Matching **two arbitrary graphs** is **NP-hard**

Is it reasonable?

$\rightarrow$ Matching **two arbitrary graphs** is **NP-hard**

$\rightarrow$ But we are **data-driven**!

Is it reasonable?

$\rightarrow$ Matching **two arbitrary graphs** is **NP-hard**

$\rightarrow$ But we are **data-driven**!

$\rightarrow$ For a given **dataset** a solver might exist!

Is it reasonable?

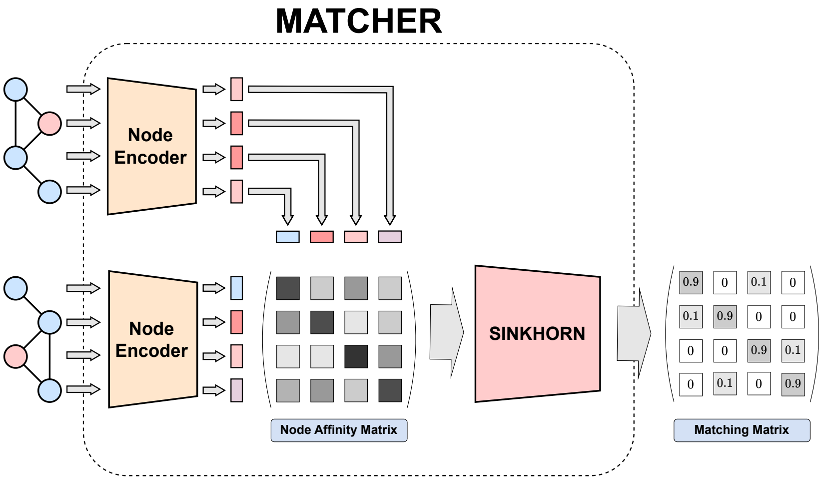
$\rightarrow$ Matching **two arbitrary graphs** is **NP-hard**

$\rightarrow$ But we are **data-driven**!

$\rightarrow$ For a given **dataset** a solver might exist!

$\rightarrow$ Let's **learn** it!

Challenges: trainable matching module



Challenges: loss choice

For $P \in \sigma_n$ permutation matrix,

$$||A - PA'P^T||_F^2 = ||AP - PA'||_F^2 = \sum_{i,j,k,l} P_{i,k} P_{j,l} d(A_{i,j}, A'_{k,l})$$

For $T \in \pi_n$ matching matrix:

$$||A - TA'T^T||_F^2 \neq ||AT - TA'||_F^2 \neq \underbrace{\sum_{i,j,k,l} T_{i,k} T_{j,l} d(A_{i,j}, A'_{k,l})}_{\mathcal{L}_{GW}(A,A',T)}$$

We need to choose a relaxation of the graph matching loss.

Theorem

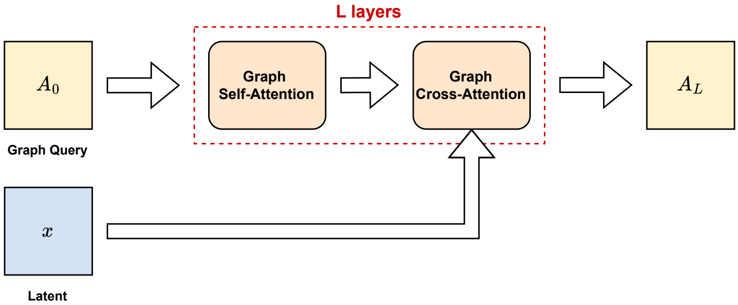
$\mathcal{L}_{GW}$ is the only relaxation such that

$$\mathcal{L}_{GW}(A, A', T) = 0 \iff \exists P \in \sigma_n, A = PA'P^T$$

Challenges: expressive decoder

- Many works on graph encoding models $x = f_\theta(A)$
- Few works on graph decoding models $A = f_\theta(x)$

High level idea: augment **Evoformer** (AlphaFold) with **cross-attention**.

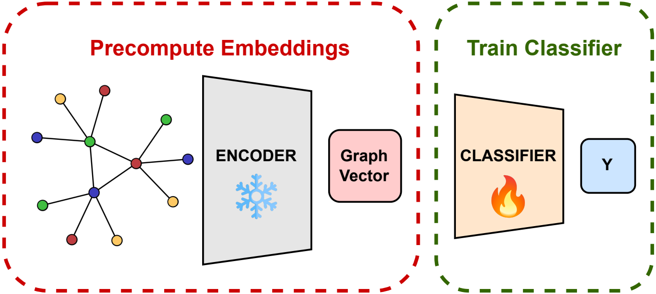


Applications:

Once GRALE is trained...

Any graph-level task can be performed in the **latent space** !

From **graph classification** ...

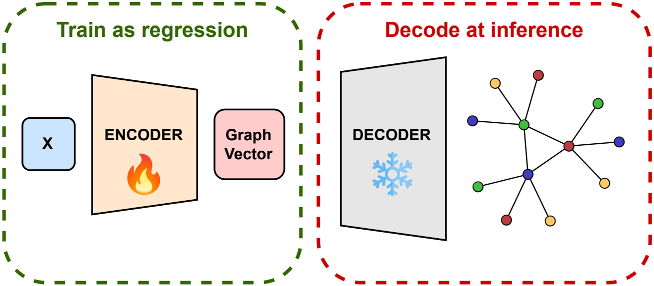


Applications:

Once GRALE is trained...

Any graph-level task can be performed in the **latent space** !

To **graph generation** ...

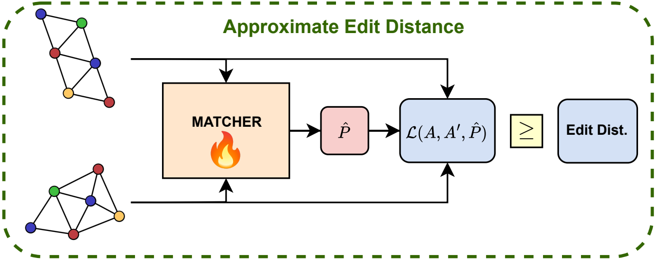


Applications:

Once GRALE is trained...

Any graph-level task can be performed in the **latent space** !

To **graph matching** ...

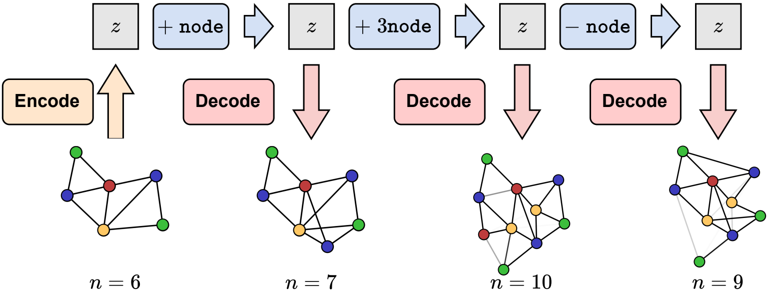


Applications:

Once GRALE is trained...

Any graph-level task can be performed in the **latent space** !

To **graph edition** ...



Applications:

Once GRALE is trained...

Any graph-level task can be performed in the **latent space** !

To **graph interpolation** ...