Efficient Graph Similarity Computation with Alignment Regularization

Wei Zhuo   Guang Tan

Shenzhen Campus of Sun Yat-sen University

NEURAL INFORMATION PROCESSING SYSTEMS

Graph Edit Distance

GED as a graph similarity measure has been popularly adopted on graph search queries, due to its capacity to capture the structural and feature differences between graphs. As shown in Figure 1, GED is defined as the number of edit operations in the optimal path that transforms G_i into G_j, where the possible edit operations under consideration include edge deletion/insertion, node deletion/insertion, and node relabeling.

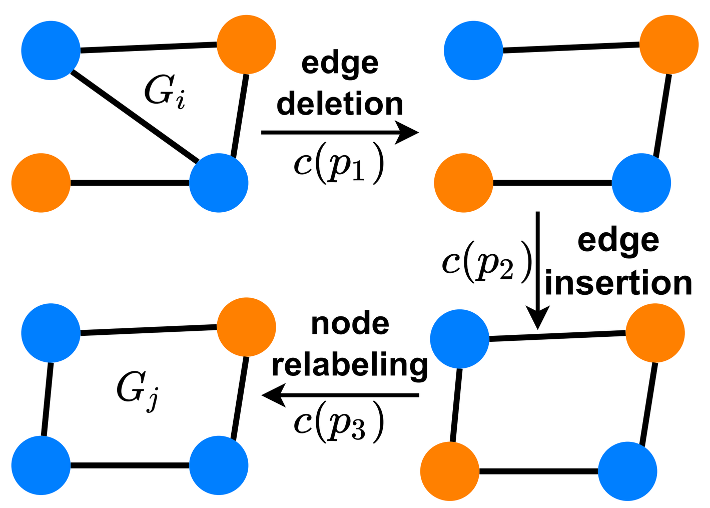
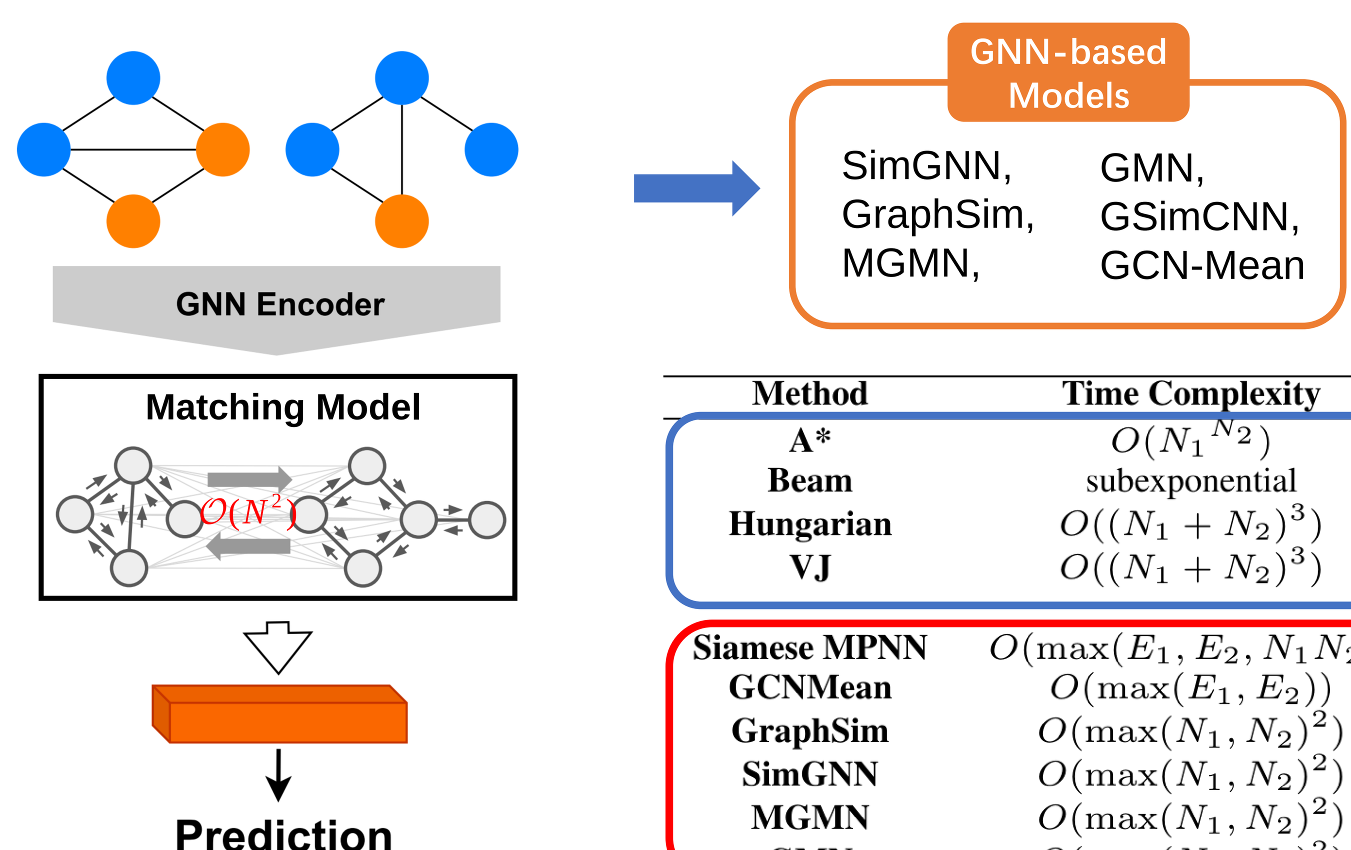


Figure 1. The optimal edit path with 3 edit operations to transform G_i to G_j. As a result, $\mathrm{GED}(G_i, G_j) = 3$.

GNN for GED Computation

With the great success of deep learning techniques, some recent works have treated GED computation as a learning problem, and use graph neural networks (GNN) for GED Computation. Such GNN-based methods have achieved higher efficiency and estimation accuracy than traditional combinatorial search-based methods.



Method	Time Complexity
A*	$O({N_1}^{N_2})$
Beam	subexponential
Hungarian	$O((N_1+N_2)^3)$
VJ	$O((N_1+N_2)^3)$
Siamese MPNN	$O(\max(E_1, E_2, N_1 N_2))$
GCNMean	$O(\max(E_1, E_2))$
GraphSim	$O(\max(N_1, N_2)^2)$
SimGNN	$O(\max(N_1, N_2)^2)$
MGMN	$O(\max(N_1, N_2)^2)$
GMN	$O(\max(N_1, N_2)^2)$

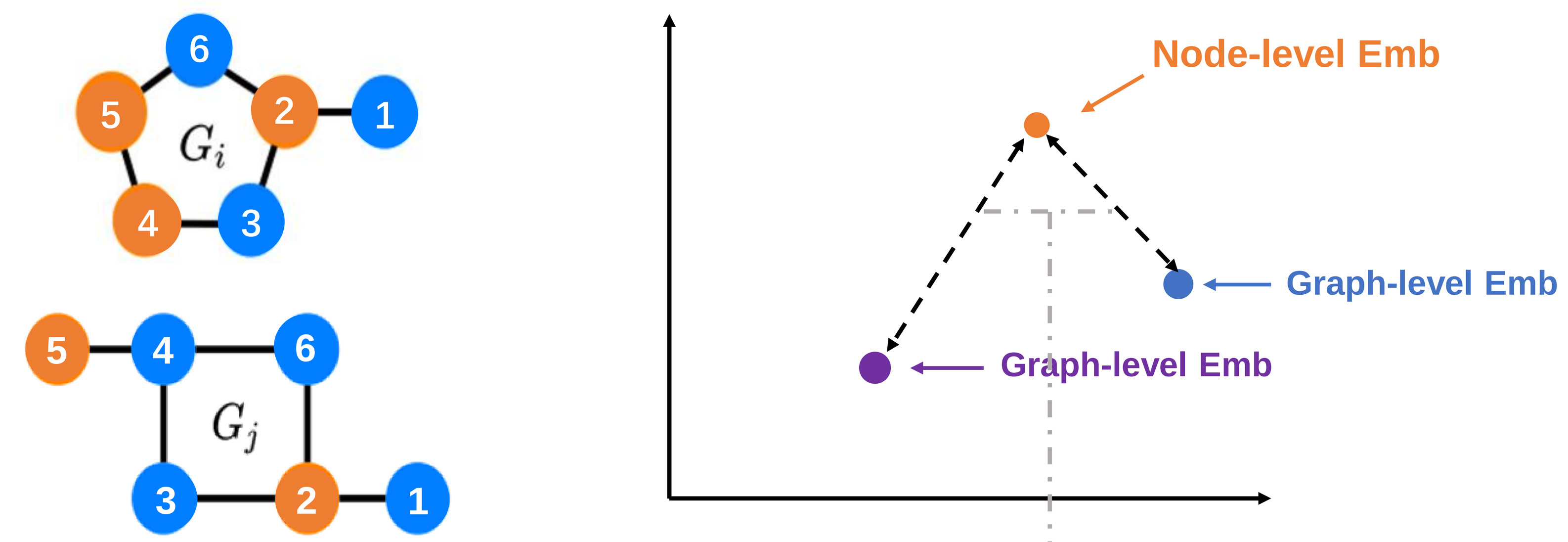
Combinatorial Search Methods (A*, Beam, Hungarian, VJ)

GNN-based Methods (Siamese MPNN, GCNMean, GraphSim, SimGNN, MGMN, GMN)

Figure 2. General model framework of GNN-based GED computation models.

The matching model captures the cross-graph node level interactions. The computational cost of such a sequential framework mainly comes from the matching model, which requires computational and memory cost quadratic in the number of nodes. Therefore, such matching model makes the learning process inefficient, especially in the inference stage.

Analyzing GED in Embedding Space



Alignment Regularization

$$\min \boxed{\mathcal{L}_{\mathrm{AReg}}} = \frac{1}{L}\sum_{\ell}^{L}\left(\boxed{\gamma_i^{(\ell)}} + \gamma_j^{(\ell)} + \left\|\gamma_i^{(\ell)} - \gamma_j^{(\ell)}\right\|_2\right)$$

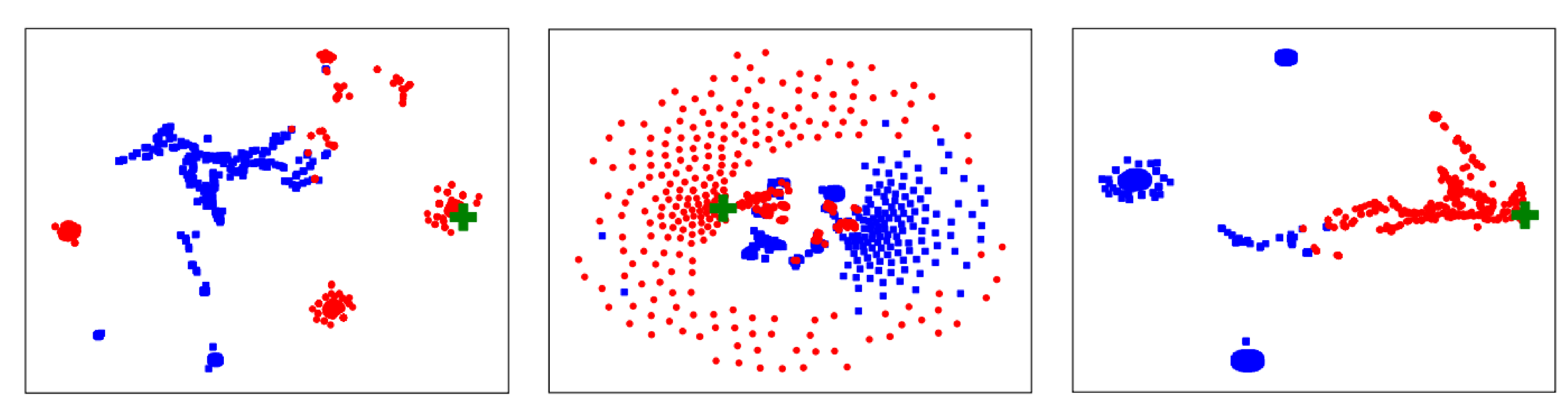
$$\gamma_i = \sum_{k}^{N} \|\mathrm{DIST}\,(\underbrace{f_\theta(\boldsymbol{A}_i[k])}_{\text{Node Level}}, \underbrace{g_\phi(\boldsymbol{A}_i)}_{\text{Graph Level}}) - \mathrm{DIST}\,(f_\theta(\boldsymbol{A}_i[k]), g_\phi(\pi(\boldsymbol{A}_j)))\|_2$$

$$f_\theta^{(\ell)}(\mathbf{A}_i[k]) = \mathrm{MLP}_\theta^{(\ell)}\left(\left(1+\xi^{(\ell)}\right)\mathbf{H}_i^{(\ell-1)}[k] + \mathbf{A}_i[k]\mathbf{H}_i^{(\ell-1)}\right) \quad g_\phi^{(\ell)}(\mathbf{A}_i) = \mathrm{MLP}_\phi^{(\ell)}\left(\sum_k^N f_\theta^{(\ell)}(\mathbf{A}_i[k])\right)$$

The best matching between two graphs can be inferred by minimizing the difference between the intra-graph node-graph similarity and cross-graph node-graph similarity.

Multi-Scale GED Discriminator

A single GED discriminator does not fully capture the dissimilarity between two graphs, while diverse discriminators may provide complementary information to reflect GED more accurately. By adaptively combining NTN and ℓ_2 distance, our model makes similar graphs closely located around the query, while dissimilar graphs far away from the query.



(a) NTN (b) ℓ_2 distance (c) NTN + ℓ_2 distance

$$\mathcal{L} = \frac{1}{n}\sum_{(G_i,G_j)\in\mathcal{D}\times\mathcal{D}} \mathrm{MSE}\left(\boxed{\alpha s_{\mathrm{NTN}}(G_i,G_j) + \beta s_p(G_i,G_j)}, \mathbf{S}_{ij}\right) + \boxed{\lambda\mathcal{L}_{\mathrm{AReg}}}$$

Multi-Scale GED Discriminator

Alignment Regularization (ignore during inference)

Efficient g**R**aph s**I**milarity **C**omputation (ERIC)

Empirical Analysis

Evaluation on benchmarks

	AIDS700					LINUX				
	mse ($\times 10^{-3}$) $\downarrow$	$\rho\uparrow$	$\tau\uparrow$	$p@10\uparrow$	$p@20\uparrow$	mse ($\times 10^{-3}$) $\downarrow$	$\rho\uparrow$	$\tau\uparrow$	$p@10\uparrow$	$p@20\uparrow$
Beam	12.090	0.609	0.463	0.481	0.493	9.268	0.827	0.714	0.973	0.924
VJ	29.157	0.517	0.383	0.310	0.345	63.86	0.581	0.450	0.287	0.251
Hungarian	25.296	0.510	0.378	0.360	0.392	29.81	0.638	0.517	0.913	0.836
SimGNN	1.573	0.835	0.678	0.417	0.489	2.479	0.912	0.791	0.635	0.650
GraphSim	2.014	0.839	0.662	0.401	0.499	0.762	0.953	0.882	0.956	0.951
GMN	4.610	0.672	0.497	0.200	0.263	2.571	0.906	0.763	0.888	0.856
EGSC	1.676	0.888	0.723	0.604	0.708	0.214	0.984	0.897	0.987	0.989
MGMN	2.297	0.904	0.736	0.456	0.534	2.040	0.965	0.858	0.956	0.920
ERIC	**1.383**	**0.906**	**0.740**	**0.679**	**0.746**	**0.113**	**0.988**	**0.908**	**0.994**	**0.996**

	IMDB					NCI109				
	mse ($\times 10^{-3}$) $\downarrow$	$\rho\uparrow$	$\tau\uparrow$	$p@10\uparrow$	$p@20\uparrow$	mse ($\times 10^{-3}$) $\downarrow$	$\rho\uparrow$	$\tau\uparrow$	$p@10\uparrow$	$p@20\uparrow$
Beam	-	-	-	-	-	-	-	-	-	-
VJ	-	-	-	-	-	-	-	-	-	-
Hungarian	-	-	-	-	-	-	-	-	-	-
SimGNN	1.437	0.871	0.752	0.710	0.769	7.767	0.576	0.435	0.023	0.040
GraphSim	1.924	0.825	0.821	0.813	0.825	8.752	0.557	0.497	0.086	0.032
GMN	4.320	0.665	0.601	0.588	0.593	11.710	0.336	0.358	0.017	0.019
EGSC	0.573	**0.939**	**0.829**	0.872	0.883	9.356	0.545	0.414	0.055	0.078
MGMN	0.496	0.881	0.803	0.874	0.861	9.631	0.492	0.426	0.015	0.051
ERIC	**0.385**	0.890	0.791	**0.882**	**0.891**	**7.127**	**0.591**	**0.525**	**0.118**	**0.080**

Inference time comparison

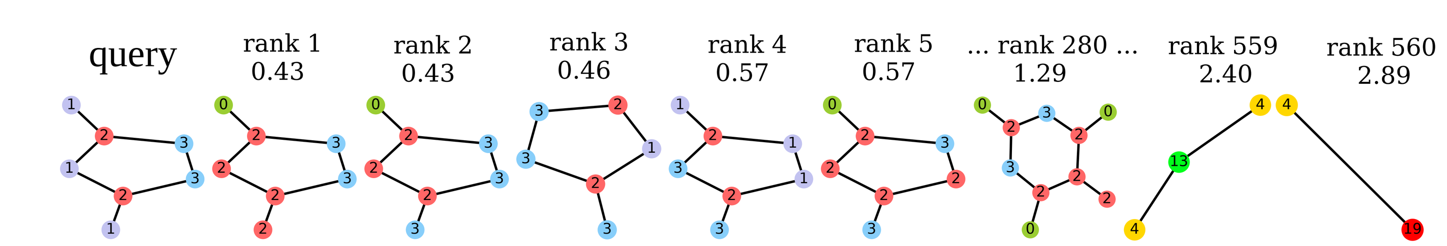
Dataset	SimGNN	GraphSim	GMN	MGMN	EGSC	ERIC
AIDS700	10.773	14.043	23.975	11.337	8.763	**6.662**
LINUX	19.347	31.238	82.489	22.574	21.573	**18.969**
IMDB	225.682	379.480	1253.551	357.933	133.437	**48.750**
NCI109	2913.178	3463.620	$> 10^4$	3726.834	2097.405	**1763.356**

Transferability

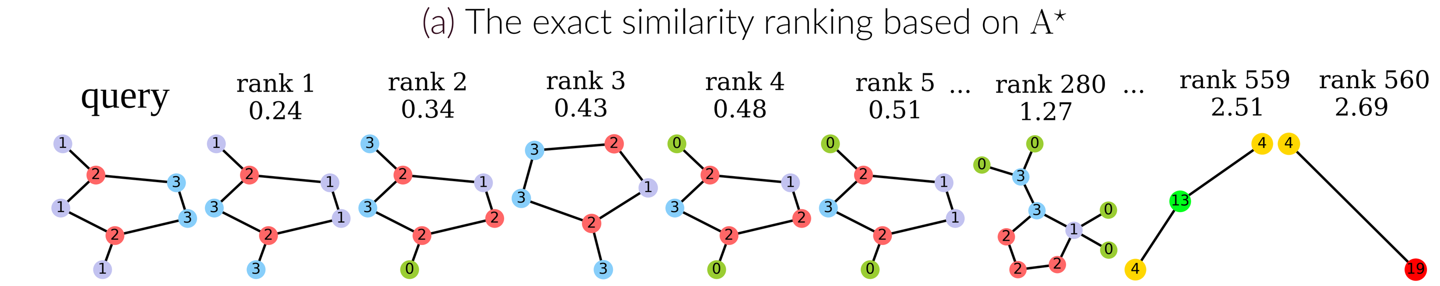
	AIDS700			LINUX		
	mse	ρ	$p@10$	mse	ρ	$p@10$
SimGNN	1.573	0.835	0.417	2.479	0.912	0.635
SimGNN+AReg	**1.439**	**0.858**	**0.506**	**1.974**	**0.945**	**0.658**
EGSC	1.676	0.888	0.604	0.214	0.984	0.987
EGSC+AReg	**1.478**	**0.904**	**0.643**	**0.142**	**0.989**	**0.992**

- ERIC consistently achieve state-of-the-arts performance across all evaluation metric.
- Alignment Regularization can be incorporated into existing methods and improve their performance, such as SimGNN and EGSC.
- ERIC is faster than all baseline models in the inference stage.

Visualization of Graph Search



(a) The exact similarity ranking based on A*



(b) The predicted similarity ranking based on ERIC

References

[1] Yunsheng Bai, Hao Ding, Song Bian, Ting Chen, Yizhou Sun, and Wei Wang. Simgnn: A neural network approach to fast graph similarity computation. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pages 384–392, 2019.