Università della Calabria
Department of Mathematics and Computer Science

Approximating Graph Edit Distance through Graph Neural Networks: Methods, Limitations, and Proposals

Master's Degree in Artificial Intelligence and Computer Science

Supervisors
Ch.mo. Prof Giorgio Terracina
Dott. Sebastiano A. Piccolo

Candidate
Federico Calabrò 247976

Context and Motivation

- Graph Edit Distance (GED) is crucial in comparing graphs, useful in fields like bioinformatics, social networks, and computer vision.
- GED calculation is **NP-Hard**, meaning exact computation is infeasible for large graphs. Hence, it needs to be approximated.
- Classical approximations rely on heuristic algorithms such as A* search and graph matching techniques.
- Recently, Graph Neural Networks (GNNs) have emerged as promising methods for faster and more accurate GED computation.

Example

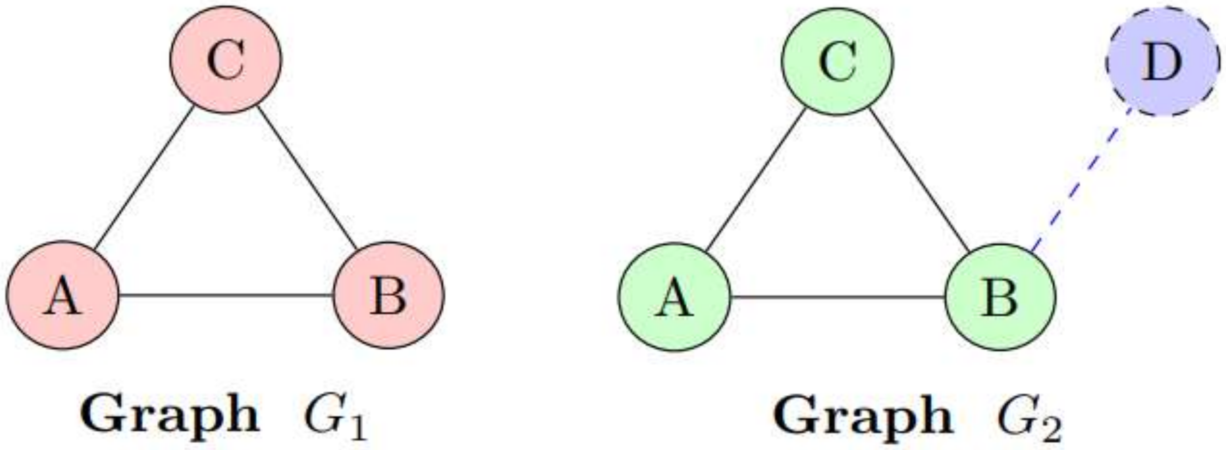


Figure 4.3: Graph Edit Distance Example.

Transforming G_1 to G_2 by adding vertex D and edge (B, D).

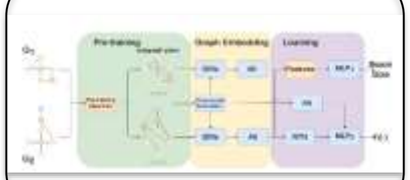
Total GED value: $1 + 1 \ = \ 2$

Contribution (1)

- A literature review was conducted, identifying **20 key papers** on GNN-based GED approximation and their trends. Four significant approaches were selected as the most representative:

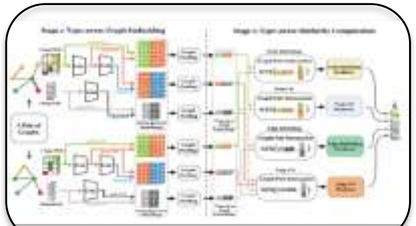
SimGNN

The foundational GNN model, introducing core features later adopted by other approaches.



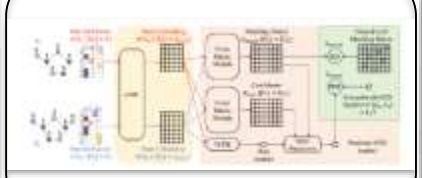
GPN

Proposing an innovative hybrid approach for GED approximation.



TaGSim

The first to re-think GED as the sum of 3 values: ged_nc, ged_in, ged_ie



GedGNN

Current state-of-the-art model in performance which combines the previous three.

2019 | 2022 | 2022 | 2023

Contribution (2)

- Development of a more flexible and robust codebase on top of GedGNN's one by enhancing its code and removing some of its limitations.
- Development of an algorithm to generate artificial datasets with graphs (pairs) at known GED distance for both training and testing.
- Set of experiments (64 combinations) showing that models lack the ability to generalize out of distribution.

Conclusion

trainset	testset	mse	mae	acc
IMDB	IMDB	0,816	**0,634**	0,58
IMDB	Linux	9,399	1,27	0,221
IMDB	1000g_100n	13,861	363,687	0,005
IMDB	Medium	201,372	4106,777	0,005
Linux	IMDB	13,153	3,539	0,075
Linux	Linux	1,161	**0,315**	0,735
Linux	1000g_100n	869,809	4367,593	0
Linux	Medium	212,754	3801,531	0
1000g_100n	IMDB	47,711	5,86	0,033
1000g_100n	Linux	28,688	**2,126**	0,105
1000g_100n	1000g_100n	0,708	78,892	0,013
1000g_100n	Medium	411,299	6885,001	0,003
Medium	IMDB	65,417	7,611	0,103
Medium	Linux	9,204	**1,17**	0,261
Medium	1000g_100n	4,651	259,151	0,003
Medium	Medium	0,718	267,837	0,003

Table 6.4: Results for GedGNN models

Models trained on specific data (IMDB and Linux) show strong performance but only within their original data distribution.

Future work: Increase the density of synthetic datasets to further improve model performance and generalization.

Thanks for your attention!!