Graph Neural Networks and Optimal Transport for Distance Learning between Graphs

Aldo Moscatelli[1], Jason Piquenot[1], Maxime Bérar[1], Pierre Héroux[1], and Sébastien Adam[1]

[1]LITIS Lab, University of Rouen Normandy,France

Abstract

Graphs are widely used data representation structures, modeling interactions between elements of a set. One challenge with graphs is that dot products and metrics are not defined in the graph space. One classical way to measure distances between graphs is the Graph Edit Distance (GED), defined as the minimum cost of a set of operations transforming one graph into another. Those operations are classically *deletion*, *insertion*, and *substitution* at edge and node levels. The mathematical resolution of this NP-complete problem is costly to compute with classical combinatorial optimization algorithms like $\mathbf{A}^*$, depth-first search[1], or Integer Linear Programs (ILP)[2] and even not feasible for large graphs. Several approaches aim to approximate the GED by relaxing constraints, as in the Bipartite Graph Edit Distance[3] or an ILP with time constraints.

In our presentation, we consider the computation of the GED between two graphs as a supervised metric learning problem, combining the representation power of Graph Neural Networks (GNNs) and optimal transport.

Indeed, GNNs are a natural approach to embedding graphs in a Euclidean space. At each layer, GNNs update the embedding of each input graph node, with an aggregation of the previous node embeddings and their local neighborhood. Therefore, iteratively, the graph structure is taken into account inside each node representation. The obtained set of node embeddings can be seen as a graph distribution in the latent representation space.

To learn an approximation of the GED in an end-to-end fashion, our model consists of two siamese GNNs and a comparison block placed at the end: each graph pair's nodes are augmented by positional encoding, then embedded by multiple Graph Isomorphism Network layers[4], the obtained embeddings are compared through a Multi-Layer Perceptron and discrete optimal transport[5][6] applied on a Euclidean metric defined in the embedding space. Optimal transport guarantees to preserve metrics properties such as symmetry, positivity, and triangular inequality. It also guarantees permutation invariant matching which is a desired property in graph representation learning. Using discrete optimal transport also enables the extraction of an edit path from one graph to another, bringing explainability to decisions taken by the model. We provide experimental results on benchmark datasets and comparisons, with recent works in the domain, with similar approaches[7][8].

References

[1] Jean-Yves Ramel Patrick Martineau. Zeina Abu-Aisheh, Romain Raveaux. An exact graph edit distance algorithm for solving pattern recognition problems. *4th International Conference on Pattern Recognition Applications and Methods 2015.*

[2] Julien Lerouge, Zeina Abu-Aisheh, Romain Raveaux, Pierre Héroux, and Sébastien Adam. Exact Graph Edit Distance Computation Using a Binary Linear Program. In *Structural, Syntactic, and Statistical Pattern Recognition 2016.*

[3] Kaspar Riesen and Horst Bunke. Approximate graph edit distance computation by means of bipartite graph matching. *Image and Vision Computing*, 27(7):950–959, 2009.

[4] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks?, 2018.

[5] Gabriel Peyré and Marco Cuturi. Computational optimal transport. 2018.

[6] Sébastien Bougleux and Luc Brun. Linear Sum Assignment with Edition. Research report, Normandie Université ; GREYC CNRS UMR 6072, March 2016.

[7] Khoa D. Doan, Saurav Manchanda, Suchismit Mahapatra, and Chandan K. Reddy. Interpretable graph similarity computation via differentiable optimal alignment of node embeddings. SIGIR, 2021.

[8] Rishabh Ranjan, Siddharth Grover, Sourav Medya, Venkatesan Chakaravarthy, Yogish Sabharwal, and Sayan Ranu. GREED: A neural framework for learning graph distance functions. NeurIPS, 2022.