GOT: An Optimal Transport framework for Graph comparison

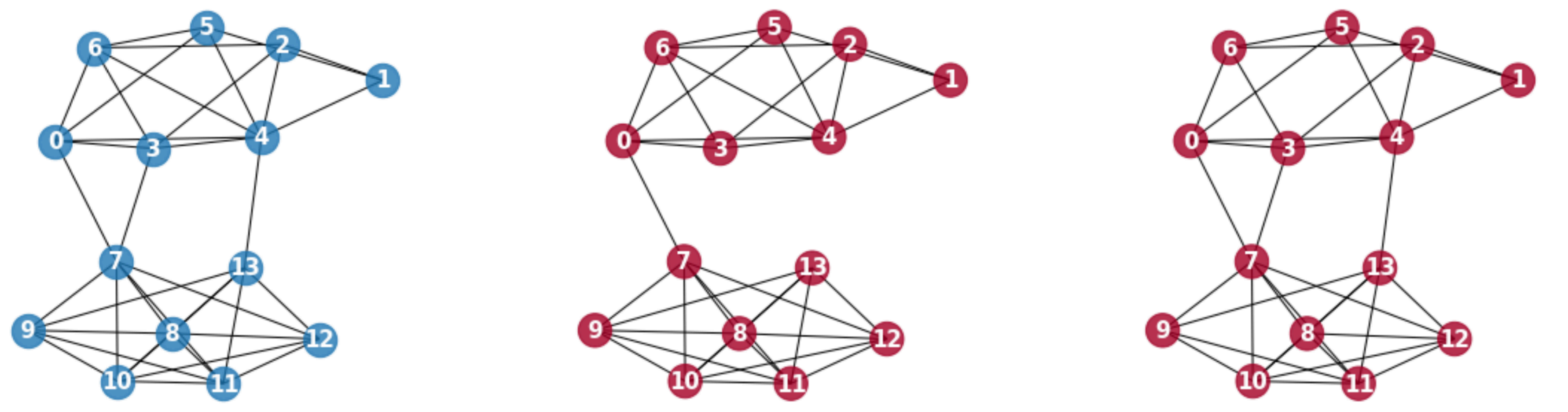
Hermina Petric Maretic, Mireille EL Gheche, Giovanni Chierchia and Pascal Frossard

Summary

- Optimal transport framework for graph comparison
- Explicit expression of the Wasserstein distance between graphs based on corresponding smooth signal distributions
- Structurally meaningful measure for comparing graphs
- Graph alignment algorithm based on the new distance
- Prediction of signal behavior on another graph through the explicit signal transportation plan

Structurally meaningful graph distance

GOT distance takes into account the global impact that a difference has on the graph structure



(a) $\mathcal{G}_1$

(b) $\mathcal{G}_2$: $\|L_1 - L_2\|_2 = 2.828$, $\mathcal{W}_2^2(\nu^{\mathcal{G}_1}, \mu^{\mathcal{G}_2}) = 0.912$

(c) $\mathcal{G}_3$: $\|L_1 - L_3\|_2 = 2.828$, $\mathcal{W}_2^2(\nu^{\mathcal{G}_1}, \mu^{\mathcal{G}_3}) = 0.013$

Graph signals and optimal transport

- Each graph defines a unique probability distribution of smooth signals

$$\nu^{\mathcal{G}_1} = \mathcal{N}(0, L_1^\dagger) \quad \mu^{\mathcal{G}_2} = \mathcal{N}(0, L_2^\dagger)$$

- Explicit expression for the graph Wasserstein distance

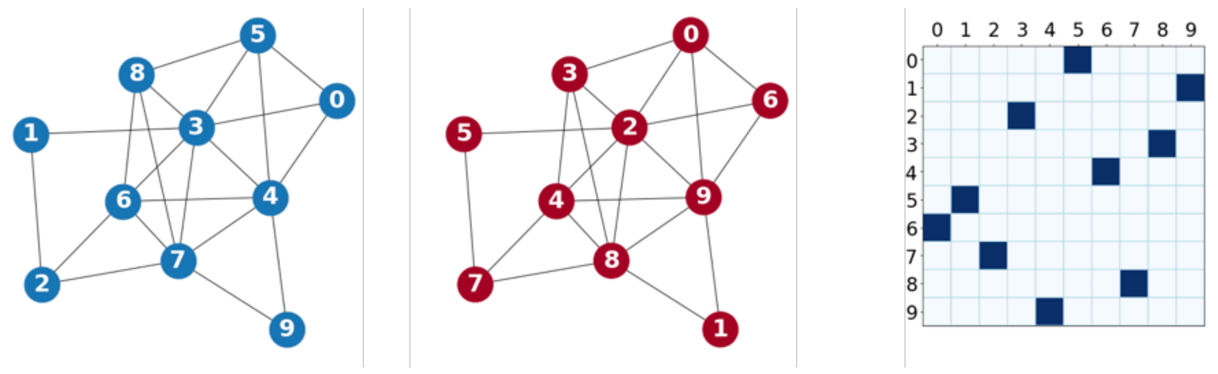
$$W_2^2(\nu^{\mathcal{G}_1}, \mu^{\mathcal{G}_2}) = \mathrm{Tr}\left(L_1^\dagger + L_2^\dagger\right) - 2\,\mathrm{Tr}\left(\sqrt{L_1^{\frac{\dagger}{2}} L_2^\dagger L_1^{\frac{\dagger}{2}}}\right)$$

- Expression for the signal transport plan

$$T(x) = L_1^{\frac{\dagger}{2}}\left(L_1^{\frac{\dagger}{2}} L_2^\dagger L_1^{\frac{\dagger}{2}}\right)^{\frac{\dagger}{2}} L_1^{\frac{\dagger}{2}} x$$

GOT Algorithm

- Alignment problem – what if we enumerate the nodes differently?



$$\mu^{\mathcal{G}_1} = \mathcal{N}(0, L_1^\dagger) \neq \mathcal{N}(0, (P^\top L_1 P)^\dagger) = \mu^{\mathcal{G}_2}$$

- The Wasserstein distance becomes

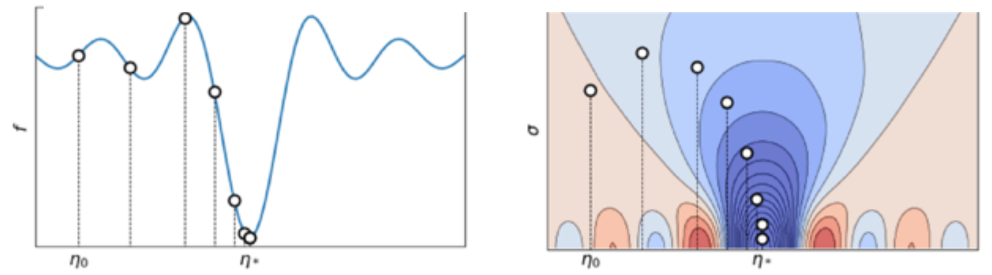
$$\mathcal{W}_2^2(\nu^{\mathcal{G}_1}, \mu_P^{\mathcal{G}_2}) = \mathrm{Tr}\left(L_1^\dagger + P^T L_2^\dagger P\right) - 2\,\mathrm{Tr}\left(\sqrt{L_1^{\frac{\dagger}{2}} P^T L_2^\dagger P L_1^{\frac{\dagger}{2}}}\right)$$

- Finding the right alignment

$$\underset{P \in \mathbb{R}^{N\times N}}{\text{minimize}} \;\; \mathcal{W}_2^2(\nu^{\mathcal{G}_1}, \mu_P^{\mathcal{G}_2}) \qquad \text{s.t.} \begin{cases} P \in [0,1]^N \\ P\mathbb{1}_N = \mathbb{1}_N \\ \mathbb{1}_N^\top P = \mathbb{1}_N \\ P^\top P = \mathrm{I}_{N\times N}, \end{cases}$$

- Stochastic gradient descent

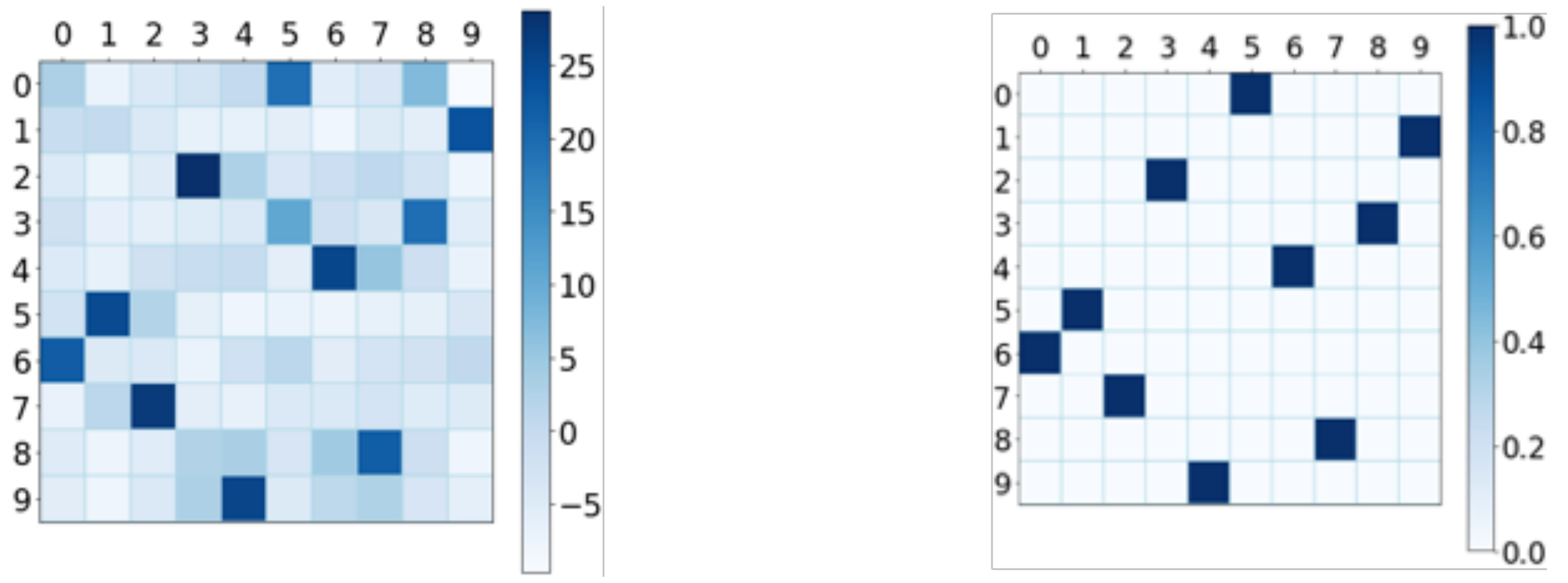
$$\underset{\theta}{\text{minimize}} \;\; \mathbb{E}_{P\sim q_\theta}\left\{\mathcal{W}_2^2(\nu^{\mathcal{G}_1}, \mu_{\mathcal{S}_\tau(P)}^{\mathcal{G}_2})\right\}$$



(a) Plot of $f(t) = -\mathrm{sinc}(t)$ (b) Contours of $J(\eta, \sigma) = \mathbb{E}_{t\sim\mathcal{N}(\eta,\sigma^2)}\{f(t)\}$

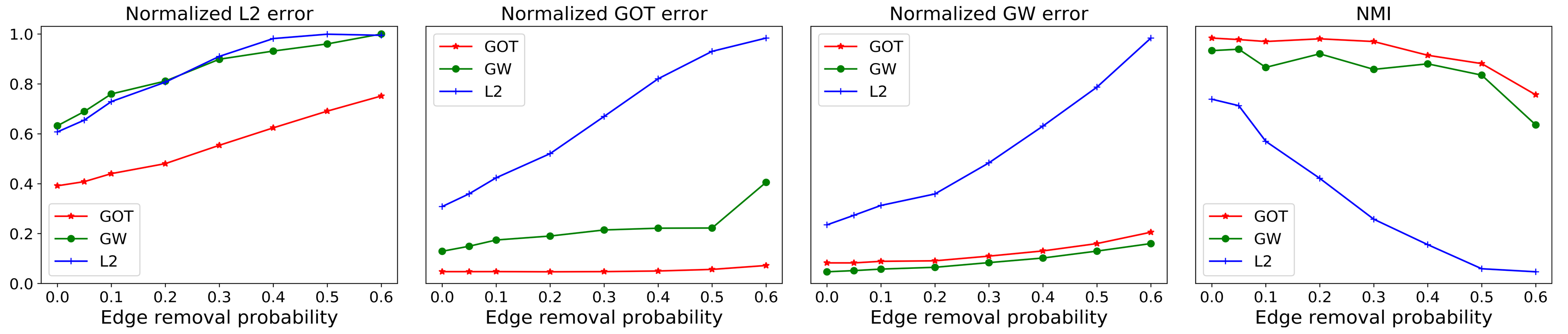
- Implicit projections with the Sinkhorn operator

$$\mathcal{S}_\tau(P) = A\exp(P/\tau)B.$$

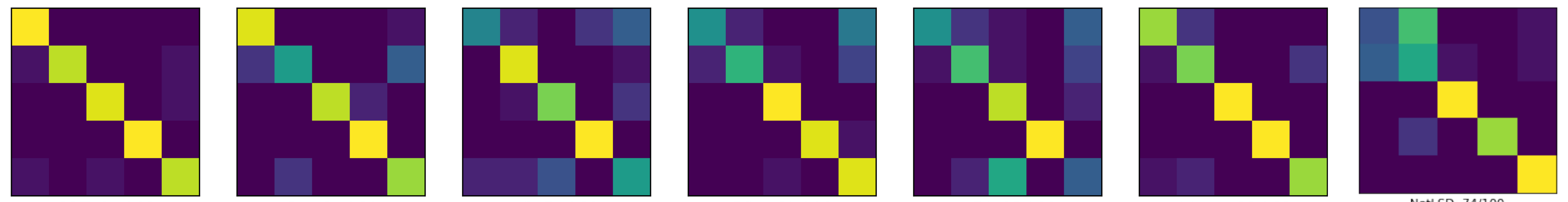


Results

- Alignment and community detection in structured graphs
  - Influence of distance definition on alignment recover
  - Comparison in terms of all three distances and community recovery NMI



- Graph classification
  - Instances of 5 different random graph models
  - Confusion matrices for different methods



- Graph signal transportation
  - For each class of images, we construct a K-NN graph between pixels (784 nodes)
  - Each image is a signal on the corresponding graph
  - Using the transportation plan T, we predict images of other classes, while keeping the properties of a given signal