Subgraph Matching via Partial Optimal Transport

Wen-Xin Pan, Isabel Haasler, Pascal Frossard

École Polytechnique Fédérale de Lausanne (EPFL)

Signal Processing Laboratory 4 (LTS4)

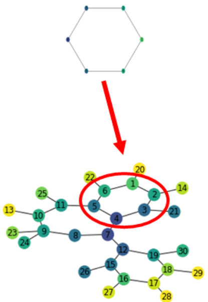
ISIT 2024

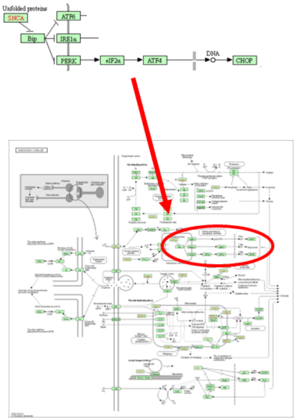
Outlines

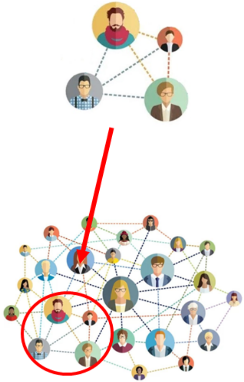
Table of Contents

Background

Search for *predefined* patterns within an object



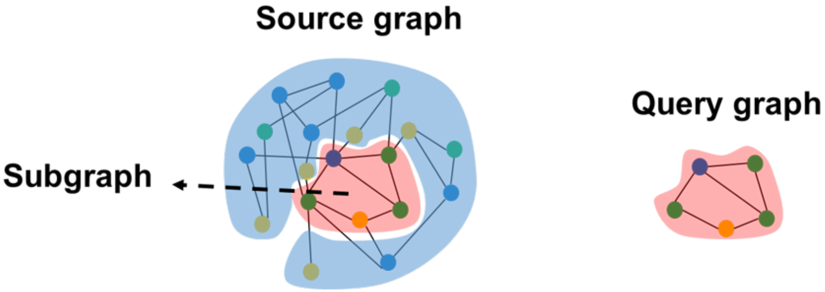




Research questions

Attributed subgraph matching, with a *predefined* query graph,

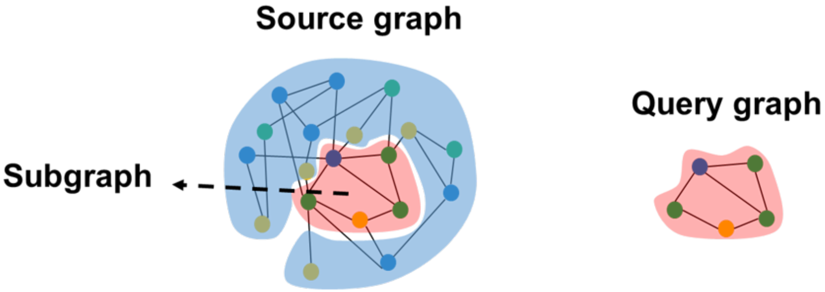
Attributes = colors, chemical elements, personal identities...



Research questions

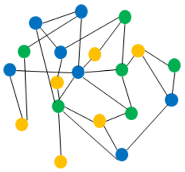
Attributed subgraph matching, with a *predefined* query graph,

Q1 Is there any subgraph? **Q2** Where? **Q3** How similar with the query?



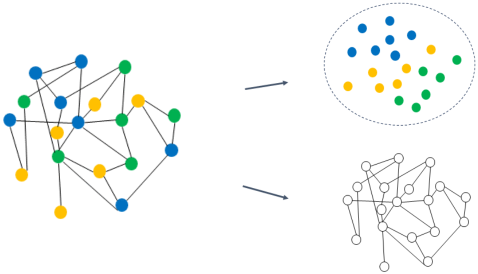
Framework

- Measure the difference between the subgraph and query
- Combine attribute and structure information



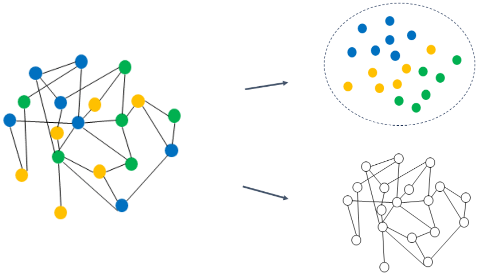
Framework

- Measure the difference between the subgraph and query
- Combine attribute and structure information



Framework

- Measure the difference between the subgraph and query
- Combine attribute and structure information

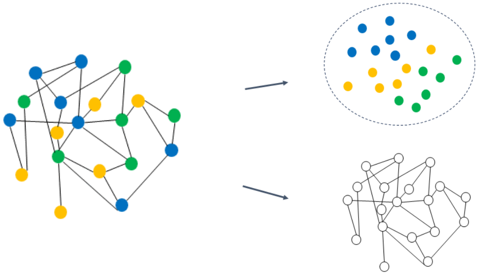


$\rightarrow$ Minimize *attribute* difference D^F

$\rightarrow$ Minimize *structure* difference D^S

Framework

- Measure the difference between the subgraph and query
- Combine attribute and structure information



$\rightarrow$ Minimize *attribute* difference D^F

$\rightarrow$ Minimize *structure* difference D^S

Minimize *total* difference $D = (1-\alpha)D^F + \alpha D^S$

Existing formulations

- **Quadratic assignment problems (QAPs)**[1]
  - aim to find a *binary* assignment matrix $\boldsymbol{X} \in \{0,1\}^{n \times n}$ between two graphs
- ! are complex *combinatorial* optimization problems
- ! have troubles with *inexact* matching problems
- **Index-based methods**[2]
  - often prioritize efficiency over accuracy
  - able tackle inexact matching
- ! typically rely on *heuristics*

[1] Lawler, "The quadratic assignment problem", Management science.
[2] Khan et al., "Nema: Fast graph search with label similarity", VLDB 2013.

Existing formulations

- **Quadratic assignment problems (QAPs)**[1]
  - aim to find a *binary* assignment matrix $\boldsymbol{X} \in \{0, 1\}^{n \times n}$ between two graphs
- ! are complex *combinatorial* optimization problems
- ! have troubles with *inexact* matching problems
- **Index-based methods**[2]
  - often prioritize efficiency over accuracy
  - able tackle inexact matching
- ! typically rely on *heuristics*

[1] Lawler, "The quadratic assignment problem", Management science.
[2] Khan et al., "Nema: Fast graph search with label similarity", VLDB 2013.

Existing formulations

- **Quadratic assignment problems (QAPs)**[1]
  - aim to find a *binary* assignment matrix $\boldsymbol{X} \in \{0,1\}^{n\times n}$ between two graphs

! are complex *combinatorial* optimization problems

! have troubles with *inexact* matching problems

- **Index-based methods**[2]
  - often prioritize efficiency over accuracy
  - able tackle inexact matching

! typically rely on *heuristics*

[1] Lawler, "The quadratic assignment problem", Management science.
[2] Khan et al., "Nema: Fast graph search with label similarity", VLDB 2013.

Existing formulations

- **Quadratic assignment problems (QAPs)**[1]
  - aim to find a *binary* assignment matrix $\boldsymbol{X} \in \{0,1\}^{n\times n}$ between two graphs

! are complex *combinatorial* optimization problems

! have troubles with *inexact* matching problems

- **Index-based methods**[2]
  - often prioritize efficiency over accuracy
  - able tackle inexact matching

! typically rely on *heuristics*

[1] Lawler, "The quadratic assignment problem", Management science.
[2] Khan et al., "Nema: Fast graph search with label similarity", VLDB 2013.

Existing formulations

- **Quadratic assignment problems (QAPs)**[1]
  - aim to find a *binary* assignment matrix $\boldsymbol{X} \in \{0,1\}^{n \times n}$ between two graphs

! are complex *combinatorial* optimization problems

! have troubles with *inexact* matching problems

- **Index-based methods**[2]
  - often prioritize efficiency over accuracy
  - able tackle inexact matching

! typically rely on *heuristics*

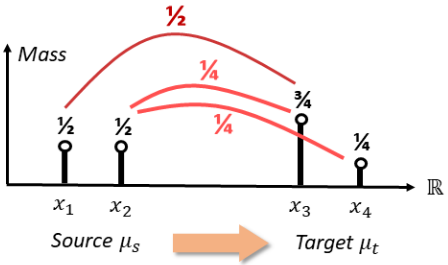
[1] Lawler, "The quadratic assignment problem", Management science.
[2] Khan et al., "Nema: Fast graph search with label similarity", VLDB 2013.

OT distances for graphs - Preview

- define *distances/metrics* for attributed graphs
- provide *efficient* matching algorithms
- able to tackle *inexact* matching

Table of Contents

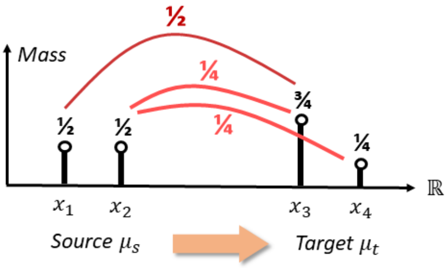
Wasserstein distance



Transport the *mass* with minimum cost.

Probability vectors $\boldsymbol{p}=\begin{bmatrix}1/2\\1/2\end{bmatrix}$ $\boldsymbol{q}=\begin{bmatrix}3/4\\1/4\end{bmatrix}$

Wasserstein distance



Transport the *mass* with minimum cost.

Probability vectors $\boldsymbol{p} = \begin{bmatrix} 1/2 \\ 1/2 \end{bmatrix}$ $\boldsymbol{q} = \begin{bmatrix} 3/4 \\ 1/4 \end{bmatrix}$

W for discrete measures

$$\mathcal{W}^{\Omega}(\mu_s, \mu_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p}, \boldsymbol{q})} \langle \boldsymbol{M}, \boldsymbol{T} \rangle \overset{\text{def.}}{=} \sum_{i,j} \boldsymbol{M}_{i,j} \boldsymbol{T}_{i,j}$$

where $\mathcal{T}(\boldsymbol{p}, \boldsymbol{q}) \overset{\text{def.}}{=} \left\{ \boldsymbol{T} \in \mathbb{R}_+^{n \times m} \mid \boldsymbol{T}\mathbf{1} = \boldsymbol{p}, \boldsymbol{T}^\top \mathbf{1} = \boldsymbol{q} \right\}$

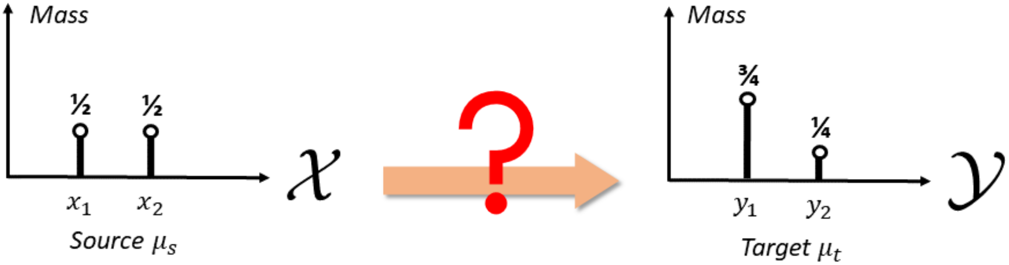
A simple cost matrix

$$\boldsymbol{M} = \begin{bmatrix} |x_1 - x_3| & |x_1 - x_4| \\ |x_2 - x_3| & |x_2 - x_4| \end{bmatrix}$$

A possible transport matrix

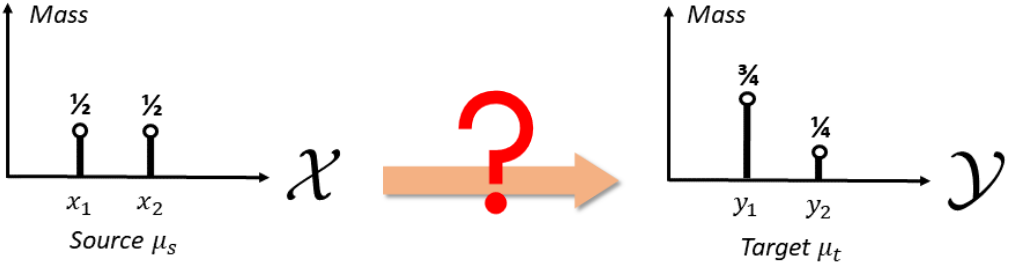
$$\boldsymbol{T} = \begin{bmatrix} 1/2 & 0 \\ 1/4 & 1/4 \end{bmatrix}$$

Gromov-Wasserstein distance



- For measures that lie in different spaces (e.g. spaces of different dimensions)

Gromov-Wasserstein distance



- For measures that lie in different spaces (e.g. spaces of different dimensions)
- The cost matrix $\boldsymbol{M}$ can not be properly defined across the spaces.

Gromov-Wasserstein distance

$$\boldsymbol{C}^s = \begin{bmatrix} 0 & |x_1 - x_2| \\ |x_2 - x_1| & 0 \end{bmatrix}$$

$$\boldsymbol{C}^t = \begin{bmatrix} 0 & |y_1 - y_2| \\ |y_2 - y_1| & 0 \end{bmatrix}$$

Gromov-Wasserstein distance

$$\boldsymbol{C}^s = \begin{bmatrix} 0 & |x_1 - x_2| \\ |x_2 - x_1| & 0 \end{bmatrix} \qquad \boldsymbol{C}^t = \begin{bmatrix} 0 & |y_1 - y_2| \\ |y_2 - y_1| & 0 \end{bmatrix}$$

$$\mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right)$$

Gromov-Wasserstein distance

For two measures that lie in possibly different spaces $\mathcal{X}$ and $\mathcal{Y}$

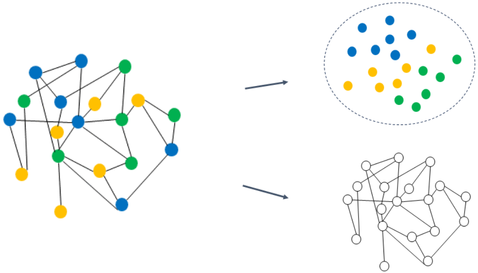
GW for discrete measures

$$\mathcal{GW}^{\mathcal{X},\mathcal{Y}}(\mu_s, \mu_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p},\boldsymbol{q})} \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

- $\mathcal{X}, \mathcal{Y}$: supports of μ_s and μ_t
- $\boldsymbol{C}^s$: cost matrix within $\mathcal{X}$
- $\boldsymbol{C}^t$: cost matrix within $\mathcal{Y}$
- $\mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right)$: loss between each elements of $\boldsymbol{C}^s$ and $\boldsymbol{C}^t$
- $\boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$: transport from point pair (i, i') to (j, j')

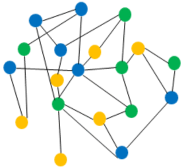
OT for attributed graphs

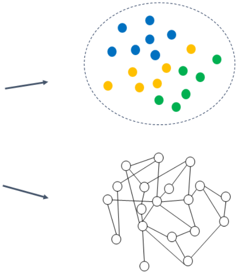
OT distances for *graphs* are defined equivalently as between *discrete measures*



OT for attributed graphs

OT distances for *graphs* are defined equivalently as between *discrete measures*





$\rightarrow$ Wasserstein distance

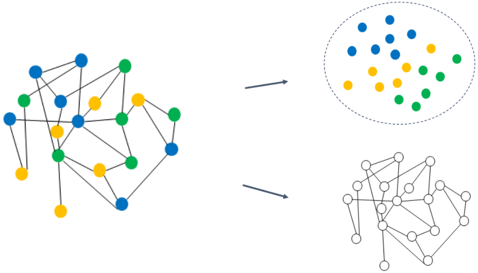
$$\min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p},\boldsymbol{q})} \langle \boldsymbol{M}, \boldsymbol{T} \rangle$$

$\rightarrow$ Gromov-Wasserstein distance

$$\min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p},\boldsymbol{q})} \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

OT for attributed graphs

OT distances for *graphs* are defined equivalently as between *discrete measures*



→ Wasserstein distance

$$\min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p}, \boldsymbol{q})} \langle \boldsymbol{M}, \boldsymbol{T} \rangle$$

→ Gromov-Wasserstein distance

$$\min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p}, \boldsymbol{q})} \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

$$\min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p}, \boldsymbol{q})} (1-\alpha) \langle \boldsymbol{M}, \boldsymbol{T} \rangle + \alpha \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

Fused Gromov-Wasserstein distance for attributed graphs[3]

For two attributed graphs G_s and G_t

FGW for attributed graphs

$$\mathcal{FGW}(G_s, G_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p},\boldsymbol{q})} (1-\alpha)\langle \boldsymbol{M}, \boldsymbol{T}\rangle + \alpha \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

- $\boldsymbol{M}$: attribute cost matrix
- $\boldsymbol{C}^s$, $\boldsymbol{C}^t$: structure cost matrices of G_s and G_t
- $\mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right)$: loss between each elements of $\boldsymbol{C}^s$ and $\boldsymbol{C}^t$
- $\boldsymbol{p}$ and $\boldsymbol{q}$: weights/importance on nodes

[3] Titouan et al., "Optimal Transport for structured data with application on graphs", ICML 2019.

$$\mathcal{FGW}(G_s, G_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p}, \boldsymbol{q})} (1-\alpha)\langle \boldsymbol{M}, \boldsymbol{T} \rangle + \alpha \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

OT distances for graphs - Details

The FGW distance

- defines a *distance/metric* between two attributed graphs
  - if with metric-typed loss functions
- is naturally a *continuous* problem
  - can be viewed as a *relaxation* of classic Quadratic Assignment Problems (QAPs)
- is a *generalization* of classic QAPs
  - allows graphs of *different sizes*

$$\mathcal{FGW}(G_s, G_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p},\boldsymbol{q})} (1-\alpha)\langle \boldsymbol{M}, \boldsymbol{T}\rangle + \alpha \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

OT distances for graphs - Details

The FGW distance

- defines a *distance/metric* between two attributed graphs
  - if with metric-typed loss functions
- is naturally a *continuous* problem
  - can be viewed as a *relaxation* of classic Quadratic Assignment Problems (QAPs)
- is a *generalization* of classic QAPs
  - allows graphs of *different sizes*

$$\mathcal{FGW}(G_s, G_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p},\boldsymbol{q})} (1-\alpha)\langle \boldsymbol{M}, \boldsymbol{T}\rangle + \alpha \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

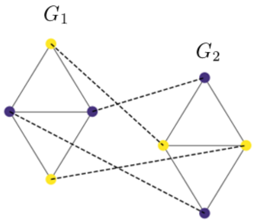
OT distances for graphs - Details

The FGW distance

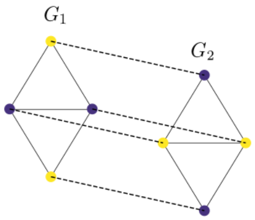
- defines a *distance/metric* between two attributed graphs
  - if with metric-typed loss functions
- is naturally a *continuous* problem
  - can be viewed as a *relaxation* of classic Quadratic Assignment Problems (QAPs)
- is a *generalization* of classic QAPs
  - allows graphs of *different sizes*

$$\mathcal{FGW}(G_s, G_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p},\boldsymbol{q})} (1-\alpha)\langle \boldsymbol{M}, \boldsymbol{T}\rangle + \alpha \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

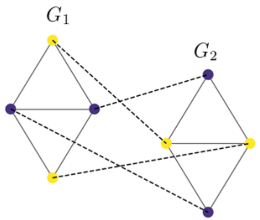
Example[4]



Wasserstein distance $= 0$



GW distance $= 0$



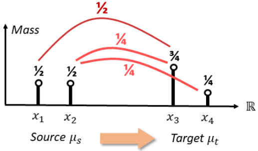
FGW distance $= 0.125$ $(\alpha = 0.5)$

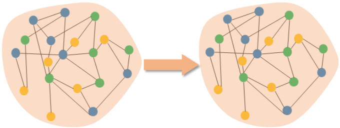
[4] Titouan et al., "Optimal Transport for structured data with application on graphs", ICML 2019.

$$\mathcal{FGW}(G_s, G_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p},\boldsymbol{q})} (1-\alpha)\langle \boldsymbol{M}, \boldsymbol{T}\rangle + \alpha \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

Partial Optimal Transport[5]

Mass is fully transported



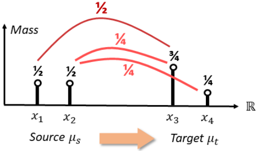


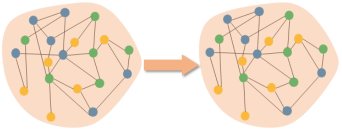
[5] Chapel, Alaya, and Gasso, "Partial optimal tranport with applications on positive-unlabeled learning", NeurIPS 2020.

$$\mathcal{FGW}(G_s, G_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p},\boldsymbol{q})} (1-\alpha)\langle \boldsymbol{M}, \boldsymbol{T}\rangle + \alpha \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

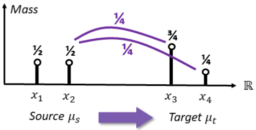
Partial Optimal Transport[5]

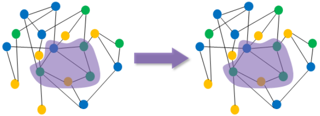
Mass is fully transported





Mass is partially transported



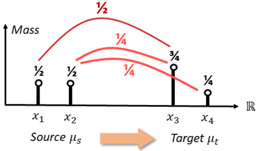


[5] Chapel, Alaya, and Gasso, "Partial optimal tranport with applications on positive-unlabeled learning", NeurIPS 2020.

$$\mathcal{FGW}(G_s, G_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p},\boldsymbol{q})} (1-\alpha)\langle \boldsymbol{M}, \boldsymbol{T}\rangle + \alpha \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j}\boldsymbol{T}_{i',j'}$$

Partial Optimal Transport[5]

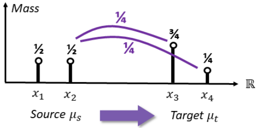
Mass is fully transported

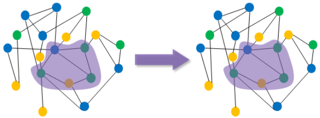


Feasible set of OT

$$\mathcal{T}(\boldsymbol{p},\boldsymbol{q}) \overset{\text{def.}}{=} \left\{ \boldsymbol{T} \in \mathbb{R}_+^{n\times m} \mid T\mathbf{1} = \boldsymbol{p}, \boldsymbol{T}^\top \mathbf{1} = \boldsymbol{q} \right\}$$

Mass is partially transported



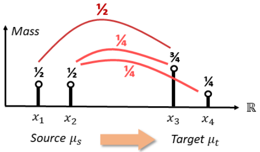


[5] Chapel, Alaya, and Gasso, "Partial optimal tranport with applications on positive-unlabeled learning", NeurIPS 2020.

$$\mathcal{FGW}(G_s, G_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p},\boldsymbol{q})} (1-\alpha)\langle \boldsymbol{M}, \boldsymbol{T}\rangle + \alpha \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j}\boldsymbol{T}_{i',j'}$$

Partial Optimal Transport[5]

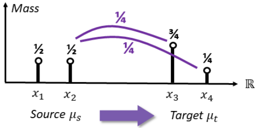
Mass is fully transported



Feasible set of OT

$$\mathcal{T}(\boldsymbol{p},\boldsymbol{q}) \stackrel{\text{def.}}{=} \left\{\boldsymbol{T} \in \mathbb{R}_+^{n\times m} \mid T\mathbf{1} = \boldsymbol{p}, \boldsymbol{T}^\top \mathbf{1} = \boldsymbol{q}\right\}$$

Mass is partially transported



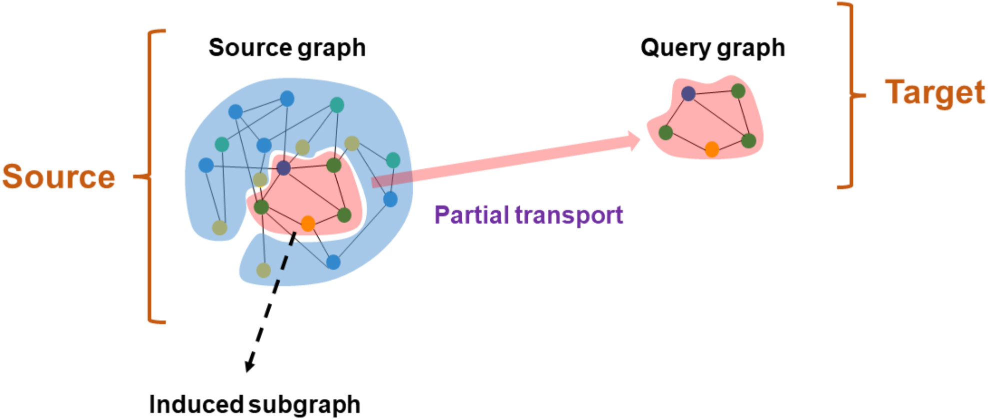
Feasible set of Partial OT

$$\mathcal{T}_s(\boldsymbol{p},\boldsymbol{q}) \stackrel{\text{def.}}{=} \left\{\boldsymbol{T} \in \mathbb{R}_+^{n\times m} \mid T\mathbf{1} \le \boldsymbol{p}, \boldsymbol{T}^\top \mathbf{1} \le \boldsymbol{q}, \mathbf{1}^\top \boldsymbol{T}\mathbf{1} = s\right\}$$

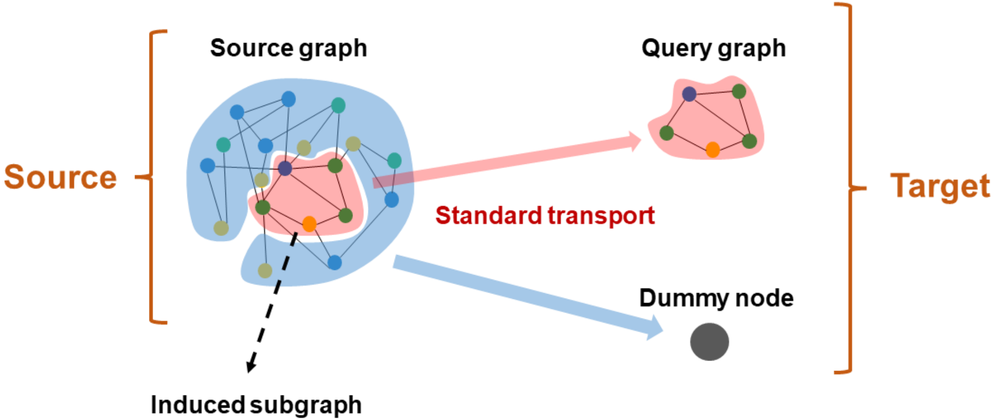
[5] Chapel, Alaya, and Gasso, "Partial optimal tranport with applications on positive-unlabeled learning", NeurIPS 2020.

Table of Contents

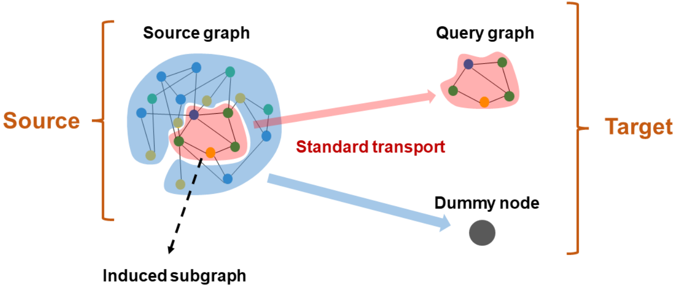
Basic algorithm



Basic algorithm



Basic algorithm



- The mass is allowed to be transported to the dummy node for free.
- The FGW distance is 0 if an exact matching is found.

$$\mathcal{FGW}(G_s, G_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p}, \boldsymbol{q})} (1-\alpha)\langle \boldsymbol{M}, \boldsymbol{T} \rangle + \alpha \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

Challenges arising in large graphs

Due to

$$\sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

- FGW distance is a *large* and *non-convex* optimization problem
- larger graphs significantly increase computation and create more local minima

Potential solution: reduce the graph sizes used in the optimization problem

$$\mathcal{FGW}(G_s, G_t) = \min_{\boldsymbol{T} \in \mathcal{T}(\boldsymbol{p},\boldsymbol{q})} (1-\alpha)\langle \boldsymbol{M}, \boldsymbol{T}\rangle + \alpha \sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

Challenges arising in large graphs

Due to

$$\sum_{i,i',j,j'} \mathcal{L}\left(\boldsymbol{C}^s_{i,i'}, \boldsymbol{C}^t_{j,j'}\right) \boldsymbol{T}_{i,j} \boldsymbol{T}_{i',j'}$$

- FGW distance is a *large* and *non-convex* optimization problem
- larger graphs significantly increase computation and create more local minima

Potential solution: reduce the graph sizes used in the optimization problem

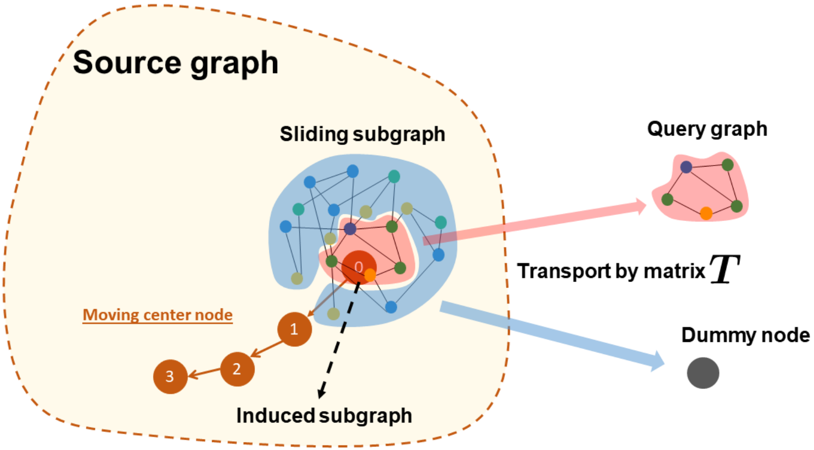
Sliding algorithm

For node in Source graph:

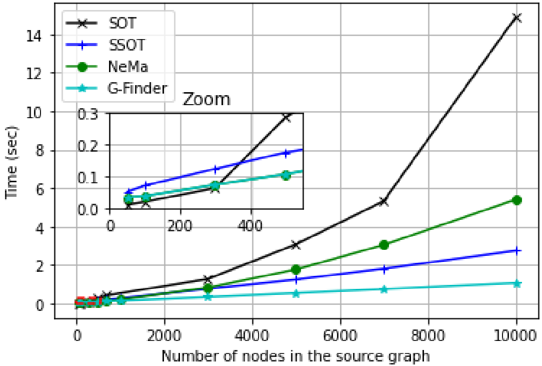
Create a sliding subgraph

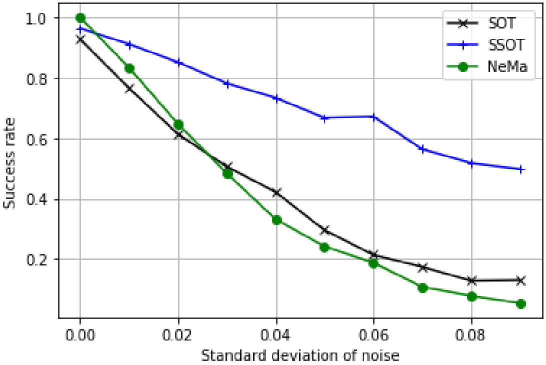
Basic algorithm (with sliding subgraph and query)

End



Experimental results





Experimental results

TABLE I: Success rate without noise.

	BZR	FIRSTMM_DB	LastFM	Deezer
SOT	1.0	0.780	1.0	–
SSOT	1.0	0.839	1.0	1.0
NeMa	1.0	0.693	1.0	0.6
G-Finder	0.9995	1.0	1.0	1.0

TABLE II: Average query time (in seconds) without noise.

	BZR	FIRSTMM_DB	LastFM	Deezer
SOT	0.005	4.351	5.659	–
SSOT	0.008	0.312	0.670	0.252
NeMa	0.027	0.384	2.550	42.229
G-Finder	0.035	0.388	3.322	21.669

TABLE III: Success rate with noise.

	BZR	FIRSTMM_DB	LastFM	Deezer
SOT	0.264	0.780	0.9	–
SSOT	0.687	0.839	1.0	1.0
NeMa	0.469	0.693	0.9	–

TABLE IV: Average query time (in seconds) with noise.

	BZR	FIRSTMM_DB	LastFM	Deezer
SOT	0.007	5.182	7.783	–
SSOT	0.090	0.322	99.763	178.976
NeMa	0.085	0.389	358.922	–

Key results from experiments

- Our methods are more robust to attribute noise compared with existing methods
- Sliding algorithm improves query efficiency and accuracy in large graphs

Table of Contents

Conclusion

- QAPs $\rightarrow$ FGW distance
- OT for probability $\rightarrow$ OT for graphs
- Partial OT $\rightarrow$ Subgraph matching
- Two subgraph matching algorithms

- Future directions
  - Entropic regularization
  - Subgraph of a different size
  - Matching in shape or image

Thank you.

References

Chapel, Laetitia, Mokhtar Z Alaya, and Gilles Gasso. "Partial optimal tranport with applications on positive-unlabeled learning". In: *Advances in Neural Information Processing Systems* 33 (2020). NeurIPS 2020, pp. 2903–2913.

Khan, Arijit et al. "Nema: Fast graph search with label similarity". In: *Proceedings of the VLDB Endowment* 6.3 (2013). VLDB 2013, pp. 181–192.

Lawler, Eugene L. "The quadratic assignment problem". In: *Management science* 9.4 (1963). Management science, pp. 586–599.

Titouan, Vayer et al. "Optimal Transport for structured data with application on graphs". In: *Proceedings of the 36th International Conference on Machine Learning*. Vol. 97. Proceedings of Machine Learning Research. ICML 2019. PMLR, 2019, pp. 6275–6284.

General formulations

Wasserstein distance

$$\mathcal{W}^{\Omega}(\mu_s, \mu_t) = \inf_{\pi \in \Pi(\mu_s, \mu_t)} \int_{\mathcal{A} \times \mathcal{B}} d_{\Omega}(a, b) d\pi(a, b)$$

where

$$\Pi(\mu_s, \mu_t) = \left\{ \pi(a, b) \geq 0, \int_{\mathcal{B}} \pi(a, b) db = \mu_s, \int_{\mathcal{A}} \pi(a, b) da = \mu_t \right\}$$

- $\mathcal{A}, \mathcal{B}$: supports of μ_s and μ_t, within the same space Ω
- $d_{\Omega}(a, b)$: loss between points within Ω
- $\pi(a, b)$: coupling between μ_s and μ_t
- $\Pi(\mu_s, \mu_t)$: all possible couplings

General formulations

Gromov-Wasserstein distance

$$\mathcal{GW}^{\mathcal{X},\mathcal{Y}}\left(\mu_s,\mu_t\right)=\inf_{\pi\in\Pi(\mu_s,\mu_t)}\int_{\mathcal{X}^2\times\mathcal{Y}^2}\mathcal{L}\left(d_{\mathcal{X}}\left(x,x'\right),d_{\mathcal{Y}}\left(y,y'\right)\right)d\pi(x,y)d\pi\left(x',y'\right)$$

- $\mathcal{X},\mathcal{Y}$: supports of μ_s and μ_t, possibly within different metric spaces
- $d_{\mathcal{X}}(x,x')$: loss within $\mathcal{X}$
- $d_{\mathcal{Y}}(y,y')$: loss within $\mathcal{Y}$
- $\pi(x,y)\pi(x',y')$: coupling between point pairs (x,x') and (y,y')

A few words about α

- For exact matching,
  (including matching in noisy datasets)
  - any $\alpha \in (0, 1)$ will work in theory
  - in practice, optimal α('s) can be found with the best success rate for a specific dataset and query graph
- For inexact matching,
  - α manually balances the importance of attributes and structure
  - there is no optimal α