A Geometric Objective for Graph Coarsening with Machine Learning Applications

Jie Chen

MIT-IBM Watson AI Lab, IBM Research

Jointly with Yifan Chen (HKBU), Rentian Yao (UIUC), Yun Yang (UIUC),
Zechen Zhang (UMN), and Yousef Saad (UMN)

Presented at SIAM Conference on Applied Linear Algebra, May 13, 2024

Introduction: Graph coarsening

Graph coarsening is a remarkably useful and ubiquitous tool in scientific computing; it is now just starting to have a similar impact in machine learning.

Main idea:

- Given a graph G, find a smaller graph $G^{(c)}$ such that it is "a good approximation" of G
- In some applications, directly obtain the solution by using $G^{(c)}$; in other applications, interpolate the solution on $G^{(c)}$ back to G
- Can do coarsening (and back-interpolation) recursively, resulting in multilevel methods

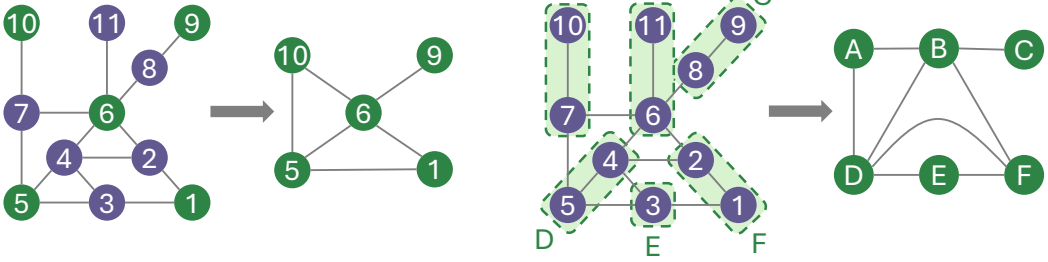
Related concepts and applications: graph partitioning; algebraic multigrid; ILU preconditioner

Synonyms: graph reduction; graph down-sampling; graph clustering; graph summarization; graph compression; graph sketching; graph pooling

Existing coarsening approaches

Two types:

- Coarse node-based; e.g., independent-set coarsening, Kron reduction (1939)
- Clustering-based; e.g., heavy edge matching, algebraic distance, spectral coarsening



What is a good coarsening?

Many coarsening methods are heuristic (e.g., based on "strength of connection") without an objective.

Spectral coarsening sets up an objective: Preserve the graph spectrum. See, e.g., Andreas Loukas's blog post "Demystifying graph coarsening".

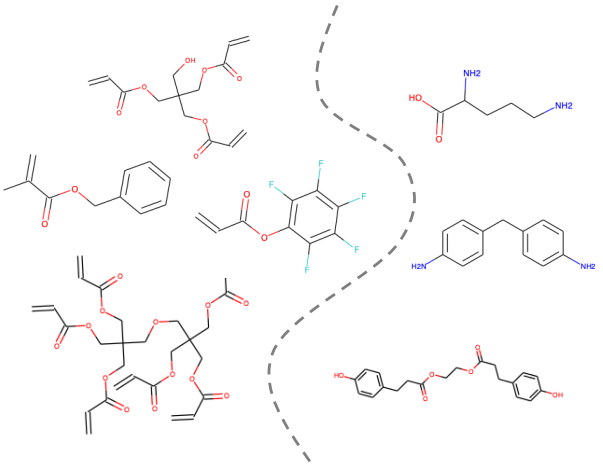
However, it is still mysterious ...

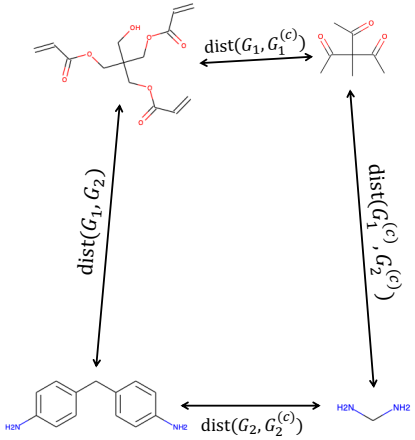
- Unclear why spectrum but not other graph properties, such as the degree distribution. (Yes, I can hear the shape of a drum; but ...)
- Unclear if preserving the spectrum will benefit the downstream application

In this work, we propose a geometric objective, which is intuitive for machine learning.

This objective sets up a framework that includes, but is not limited to, preserving the graph spectrum.

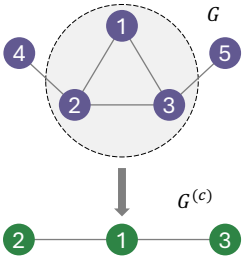
A geometric view of coarsening





Notation

- Graph G has N nodes. Node set $\mathcal{V} = \{v_i\}_{i=1}^N$
- Adjacency matrix $\mathbf{A}$; Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{A}$; normalized Laplacian $\mathcal{L} = \mathbf{D}^{-\frac{1}{2}}\mathbf{L}\mathbf{D}^{-\frac{1}{2}}$
- Assume G is connected (hence smallest eigenvalue of $\mathbf{L}$ and $\mathcal{L}$ is zero and is simple)



- The coarsened graph is denoted by $G^{(c)}$, which has $n \leq N$ nodes
- Let $\{\mathcal{V}_1, \ldots, \mathcal{V}_n\}$ be a partitioning of $\mathcal{V}$ (hence each $\mathcal{V}_k$ represents a node of $G^{(c)}$)
- Define membership matrix $\mathbf{C}_p \in \{0,1\}^{n \times N}$ with entries $\mathbf{C}_p(k,i) = \mathbf{1}(v_i \in \mathcal{V}_k)$
- Define $\mathbf{A}^{(c)} = \mathbf{C}_p \mathbf{A} \mathbf{C}_p^\top$ to be the adjacency matrix of $G^{(c)}$ (note that the diagonal of $\mathbf{A}^{(c)}$ is not necessarily empty)
- Define $\mathbf{D}^{(c)} = \mathrm{diag}(\mathbf{A}^{(c)}\mathbf{1})$. One can show that $\mathbf{D}^{(c)} - \mathbf{A}^{(c)}$ is the Laplacian of $G^{(c)}$ and $(\mathbf{D}^{(c)})^{-\frac{1}{2}}\mathbf{L}^{(c)}(\mathbf{D}^{(c)})^{-\frac{1}{2}}$ is the normalized Laplacian

$$\mathbf{C}_p = \begin{bmatrix} 1 & 1 & 1 & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix}$$

Gromov–Wasserstein distance and the metric space

A measure network is a triple $(\mathcal{X}, \mu_X, \omega_X)$:

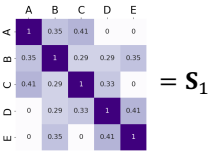
- $\mathcal{X}$ is a Polish space (a separable and completely metrizable topological space)
- μ_X is a fully supported Borel probability measure
- ω_X is a bounded measurable function on $\mathcal{X} \times \mathcal{X}$

A graph with a similarity matrix $\mathbf{S} \in \mathbb{R}^{N \times N}$ is a (finite) measure network:

- $\mathcal{X}$ is the set of graph nodes $\{v_1, \ldots, v_N\}$
- μ_X is the probably mass vector $\mathbf{m} = [m_1, \ldots, m_N]^\top \in \mathbb{R}_+^N$ with $\sum_i m_i = 1$
- ω_X is the similarity function $\omega_X(v_i, v_j) = s_{ij}$

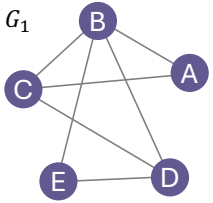
Example: $\mathbf{m}$ and $\mathbf{S}$ can be node weights and edge weights (but not necessarily).

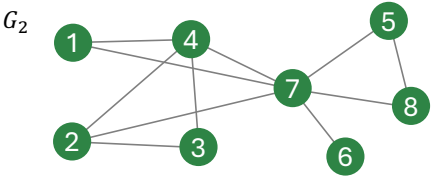
Gromov–Wasserstein distance and the metric space

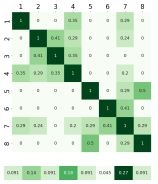


$= \mathbf{S}_1$

$= \mathbf{m}_1$

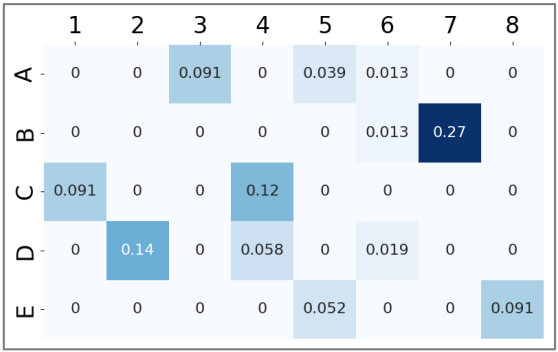






$= \mathbf{S}_2$

$= \mathbf{m}_2$



$= \mathbf{T}$

Gromov–Wasserstein distance and the metric space

Let there be two graphs G_1 and G_2.

Let $\mathbf{T} = [t_{ij}] \in \mathbb{R}^{N_1 \times N_2}$ be a transport matrix such that $\mathbf{T}\mathbf{1} = \mathbf{m}_1$ and $\mathbf{T}^\top \mathbf{1} = \mathbf{m}_2$. Let $\Pi_{1,2}$ be the set of transport matrices.

Distance between G_1 and G_2:

$$\mathrm{GW}_p(G_1, G_2)^p := \min_{\mathbf{T} \in \Pi_{1,2}} \sum_{i,j=1}^{N_1} \sum_{i',j'=1}^{N_2} |s^1_{ij} - s^2_{i'j'}|^p \mathbf{T}_{ii'} \mathbf{T}_{jj'}.$$

Rewrite:

$$\mathrm{GW}_p(G_1, G_2)^p = \min_{\mathbf{T} \in \Pi_{1,2}} \langle \mathbf{C}, \mathbf{T} \rangle \quad \text{where} \quad \mathbf{C}_{jj'} := \sum_{i,i'} |s^1_{ij} - s^2_{i'j'}|^p \mathbf{T}_{ii'}.$$

Interpretation: Assign large transport mass $\mathbf{T}_{jj'}$ to a node pair $(v^1_j, v^2_{j'})$ with small dissimilarity $\mathbf{C}_{jj'}$

One can show that GW_p is indeed a metric, modulo weak isomorphism (Chowdhury & Mémoli, 2019; Theorem 18). Therefore, GW_p induces a metric space.

Coarsening matrices

Accumulation $\mathbf{C}_p$

$\mathbf{C}_p(k,i) = \mathbf{1}(v_i \in \mathcal{V}_k)$

Example:

$$\begin{bmatrix} 1 & 1 & 1 & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix}$$

Notes:

☐ Relation to Laplacian: $\mathbf{L}^{(c)} = \mathbf{C}_p \mathbf{L} \mathbf{C}_p^\top$

Averaging $\overline{\mathbf{C}}_w$

$\overline{\mathbf{C}}_w = \mathrm{diag}(c_1^{-1}, \cdots, c_n^{-1}) \mathbf{C}_p \mathbf{M}$, $\mathbf{M} = \mathrm{diag}(\mathbf{m})$, $c_k = \sum_{v_i \in \mathcal{V}_k} m_i$

Example:

$$\begin{bmatrix} \frac{m_1}{c_1} & \frac{m_2}{c_1} & \frac{m_3}{c_1} & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix}$$

Notes:

☐ When $\mathbf{m}$ is uniform, $\overline{\mathbf{C}}_w^+ = \mathbf{C}_p^\top$

☐ We use it to define coarsened similarity matrix $\mathbf{S}^{(c)} = \overline{\mathbf{C}}_w \mathbf{S} \overline{\mathbf{C}}_w^\top$ needed by GW (consistent with the concept of semi-relaxed GW)

Projection $\mathbf{C}_w$

$\mathbf{C}_w = \mathrm{diag}(c_1^{-\frac{1}{2}}, \cdots, c_n^{-\frac{1}{2}}) \mathbf{C}_p \mathbf{M}^{\frac{1}{2}}$

Example:

$$\begin{bmatrix} \sqrt{\frac{m_1}{c_1}} & \sqrt{\frac{m_2}{c_1}} & \sqrt{\frac{m_3}{c_1}} & & \\ & & & 1 & \\ & & & & 1 \end{bmatrix}$$

Notes:

☐ $\mathbf{C}_w$ is row orthonormal

☐ $\mathbf{C}_w^\top \mathbf{C}_w = \mathbf{M}^{\frac{1}{2}} \mathbf{C}_p^\top \overline{\mathbf{C}}_w \mathbf{M}^{-\frac{1}{2}}$

☐ Relation to normalized Laplacian: $\mathcal{L}^{(c)} = \mathbf{C}_w \mathcal{L} \mathbf{C}_w^\top$ if $\mathbf{M} = \mathbf{D}/\mathrm{vol}$

☐ Later, we define $\mathbf{U} = \mathbf{M}^{\frac{1}{2}} \mathbf{S} \mathbf{M}^{\frac{1}{2}}$ and $\mathbf{U}^{(c)} = \mathbf{C}_w \mathbf{U} \mathbf{C}_w^\top$ and use their eigenvalues to bound GW

Bounding GW_2 for a single graph

Assume the similarity matrix $\mathbf{S}$ is PSD. Define for the coarse graph $\mathbf{S}^{(c)} = \overline{\mathbf{C}}_w \mathbf{S} \overline{\mathbf{C}}_w^\top$ and $\mathbf{m}^{(c)} = \mathbf{C}_p \mathbf{m}$.

Define $\mathbf{U} = \mathbf{M}^{\frac{1}{2}} \mathbf{S} \mathbf{M}^{\frac{1}{2}}$ and $\mathbf{U}^{(c)} = \mathbf{C}_w \mathbf{U} \mathbf{C}_w^\top$.

Let $\lambda_1 \geq \cdots \geq \lambda_N$ be the eigenvalues of $\mathbf{U}$ and $\lambda_1^{(c)} \geq \cdots \geq \lambda_n^{(c)}$ be the eigenvalues of $\mathbf{U}^{(c)}$.

We have

$$\mathrm{GW}_2(G^{(c)}, G)^2 \leq \lambda_{N-n+1} \underbrace{\sum_{i=1}^{n} \left(\lambda_i - \lambda_i^{(c)} \right)}_{\Delta} + \underbrace{\sum_{i=1}^{n} \lambda_i (\lambda_i - \lambda_{N-n+i}) + \sum_{i=n+1}^{N} \lambda_i^2}_{C_{\mathbf{U},n}}$$

Remarks: (1) the bound is tight when $n = N$; (2) $\Delta \geq 0$ due to the Poincaré separation theorem; (3) $C_{\mathbf{U},n} \geq 0$; (4) $C_{\mathbf{U},n}$ is independent of coarsening; (5) the choice of coarsening only affects Δ.

Bounding GW_2 for a pair of graphs

Given a pair of graphs G_1 and G_2, extend all previous notations by adding subscripts $_1$ and $_2$ respectively.

Denote by $\mathbf{T}^*$ the optimal transport plan induced by $\mathrm{GW}_2(G_1, G_2)$; and define $\mathbf{P} = \mathbf{M}_1^{-\frac{1}{2}} \mathbf{T}^* \mathbf{M}_2^{-\frac{1}{2}}$.

Define $\mathbf{V}_1 = \mathbf{P}\mathbf{M}_2^{\frac{1}{2}} \mathbf{S}_2 \mathbf{M}_2^{\frac{1}{2}} \mathbf{P}^\top$ and $\mathbf{V}_2 = \mathbf{P}^\top \mathbf{M}_1^{\frac{1}{2}} \mathbf{S}_1 \mathbf{M}_1^{\frac{1}{2}} \mathbf{P}$, both independent of coarsening.

Let $\nu_{1,1} \geq \cdots \geq \nu_{1,N_1}$ be the eigenvalues of $\mathbf{V}_1$ and $\nu_{2,1} \geq \cdots \geq \nu_{2,N_2}$ be the eigenvalues of $\mathbf{V}_2$.

We have

$$
\begin{aligned}
&|\,\mathrm{GW}_2(G_1^{(c)}, G_2^{(c)})^2 - \mathrm{GW}_2(G_1, G_2)^2| \\
&\qquad \leq \max\{\lambda_{1,N_1-n_1+1} \cdot \Delta_1 + \lambda_{2,N_2-n_2+1} \cdot \Delta_2 + C_{\mathbf{U}_1,n_1} + C_{\mathbf{U}_2,n_2}, \\
&\qquad\qquad 2\nu_{1,N_1-n_1+1} \cdot \Delta_1 + 2\nu_{2,N_2-n_2+1} \cdot \Delta_2 + 2C_{\mathbf{U}_1,\mathbf{V}_1,n_1} + 2C_{\mathbf{U}_2,\mathbf{V}_2,n_2}\}\,, \\
&\text{where } C_{\mathbf{U},\mathbf{V},n} = \textstyle\sum_{i=1}^{n} \lambda_i(\nu_i - \nu_{N-i+1}) + \sum_{i=n+1}^{N} \lambda_i \nu_i \geq 0.
\end{aligned}
$$

Remarks: (1) Even when $G_1 = G_2$, the bound can be nonzero if the coarsened graphs do not match.
(2) Δ_1 and Δ_2 are decoupled; therefore, it suffices to optimize the coarsening for each graph independently.

Interpretation: Optimizing coarsening $\Leftrightarrow$ minimizing Δ

$$\Delta + \underbrace{\sum_{i=n+1}^{N} \lambda_i}_{\perp\text{coarsening}} = \operatorname{Tr}(\mathbf{U} - \mathbf{C}_w^\top \underbrace{\mathbf{C}_w \mathbf{U} \mathbf{C}_w^\top}_{\mathbf{U}^{(c)}} \mathbf{C}_w)$$

$$= \sum_k \sum_{v_i \in \mathcal{V}_k} m_i \|\boldsymbol{\phi}_i - \boldsymbol{\mu}_k\|^2 \quad \text{with} \quad \boldsymbol{\mu}_k = \sum_{v_i \in \mathcal{V}_k} \frac{m_i}{c_k} \boldsymbol{\phi}_i$$

where $\mathbf{S}_{ij} = \langle \boldsymbol{\phi}_i, \boldsymbol{\phi}_j \rangle$ is the inner produce of the RKHS and $\|\cdot\|$ is induced by the inner product $\langle \cdot, \cdot \rangle$.

This is nothing but the objective of weighted kernel K-means clustering!

Coarsening algorithm

1: Given a kernel matrix $\mathbf{S} \in \mathbb{R}^{N \times N}$ and a probability mass vector $\mathbf{m} \in \mathbb{R}^N$
2: Start with an initial partitioning $\{\mathcal{V}_1, \ldots, \mathcal{V}_n\}$
3: **loop** until convergence
4: **for** each node v_i **do**
5: Update partition membership

$$\operatorname*{argmin}_k \operatorname{dist}(v_i, \mathcal{V}_k)^2 = \cancel{m_i} \, \|\boldsymbol{\phi}_i - \boldsymbol{\mu}_k\|^2 = \mathbf{S}_{ii} - 2 \sum_{v_j \in \mathcal{V}_k} \frac{m_j}{c_k} \mathbf{S}_{ji} + \sum_{v_{j_1}, v_{j_2} \in \mathcal{V}_k} \frac{m_{j_1} m_{j_2}}{c_k^2} \mathbf{S}_{j_1 j_2}$$

6: **end for**
7: Form a new partitioning based on the result of the above for-loop
8: // New centroids $\boldsymbol{\mu}_k$ do not need to be explicitly computed
9: **end loop**
10: **return** the coarsened adjacency matrix $\mathbf{A}^{(c)} = \mathbf{C}_p \mathbf{A} \mathbf{C}_p^\top$ (can replace $\mathbf{C}_p$ with $\mathbf{C}_w$ or $\overline{\mathbf{C}}_w$ if edge weights do not matter)

How to set $\mathbf{S}$ and $\mathbf{m}$?

- $\mathbf{S}$ must be PSD for the bounds to hold
- When $\mathbf{S}$ is the normalized Laplacian and $\mathbf{m}$ is proportional to node degrees, $\mathbf{U}$ is proportional to the Laplacian
- There are two problems in this choice:
  - First, $\mathbf{S}$ is sparse and the off-diagonal nonzero entries are negative. Is a "zero" node pair more similar than a "negative" node pair?
  - Second, it lures one to look for solutions toward the top eigenvectors of the Laplacian, which is opposite to the intuition of spectral methods.
- Hence, instead, we let $\mathbf{S}$ be the normalized signless Laplacian $\mathbf{I} + \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$. Then, $\mathbf{U}$ is proportional to the signless Laplacian $\mathbf{D} + \mathbf{A}$
- Conceptually, either "top eigenvectors of $\mathbf{A}$" or "bottom eigenvectors of $\mathbf{L}$"

Experiments

Eight datasets; average $\mathcal{V}$ is between 14 and 53; average $\mathcal{E}$ is between 17 and 199.

Compared coarsening methods (from Loukas (2019); Jin et al., (2020)):
(1) Variation Neighborhood Graph Coarsening (VNGC); (2) Variation Edge Graph Coarsening (VEGC);
(3) Multilevel Graph Coarsening (MGC); (4) Spectral Graph Coarsening (SGC).

Our method is called Kernel Graph Coarsening (KGC), initialized with K-means++.
Variant: KGC(A), which uses the best-performing baseline to initialize KGC.

$c = n/N$ denotes coarsening ratio.

Table: $GW_2(G^{(c)}, G)^2$, averaged over all graphs in PTC

Methods	$c = 0.3$	$c = 0.5$	$c = 0.7$	$c = 0.9$
VNGC	0.05558	0.04880	0.03781	0.03326
VEGC	0.03064	0.02352	0.01614	0.00927
MGC	0.05290	0.04360	0.02635	0.00598
SGC	0.03886	0.03396	0.02309	0.00584
KGC	0.03332	0.02369	**0.01255**	**0.00282**
KGC(A)	**0.03055**	**0.02346**	0.01609	0.00392

Table: Bound gap, averaged over all graphs in PTC

Methods	$c = 0.3$	$c = 0.5$	$c = 0.7$	$c = 0.9$
VNGC	0.06701	0.06671	0.05393	0.04669
VEGC	0.06246	0.06129	0.04424	0.02577
MGC	0.03203	0.03200	0.02167	0.00540
SGC	0.04599	0.04156	0.02488	0.00554
KGC	0.05145	0.05173	0.03530	0.00852
KGC(A)	0.06519	0.06402	0.04702	0.00372

Table: $|\,\mathrm{GW}_2(G_s^{(c)}, G_t^{(c)})^2 - \mathrm{GW}_2(G_s, G_t)^2|$, averaged over all s, t pairs in PTC. $c = \frac{1}{\log N_{\max}}$

Methods	**Dist. Diff.**	**Time**
VNGC	17.34 $\pm$ 0.01	6.55 $\pm$ 0.18
VEGC	9.22 $\pm$ 0.02	3.75 $\pm$ 0.01
MGC	5.31 $\pm$ 0.00	6.59 $\pm$ 0.02
SGC	6.06 $\pm$ 0.02	28.06 $\pm$ 0.10
KGC	**4.45** $\pm$ **0.03**	1.34 $\pm$ 0.33
KGC(A)	5.28 $\pm$ 0.00	**0.27** $\pm$ **0.00**

Figure: Coarsening time (left) and spectrum difference $\frac{1}{5}\sum_{i=1}^{5} \frac{\lambda_i - \lambda_i^{(c)}}{\lambda_i}$ (right) on Tumblr

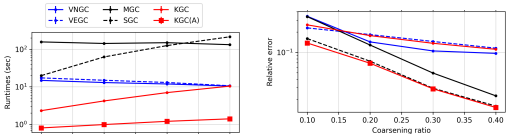


Table: Graph classification accuracy on six datasets. Method: NetLSD. $c = 0.2$

Methods	MUTAG	PTC	PROTEINS	MSRC	IMDB	Tumblr
VNGC	76.11 ± 2.25	56.69 ± 2.52	65.44 ± 1.57	14.92 ± 1.57	53.90 ± 0.50	50.43 ± 2.62
VEGC	84.59 ± 2.02	56.39 ± 2.03	64.08 ± 1.11	16.80 ± 2.15	64.20 ± 1.90	48.26 ± 1.71
MGC	84.15 ± 3.14	54.66 ± 3.59	66.16 ± 1.64	15.36 ± 1.80	69.50 ± 1.42	50.14 ± 2.67
SGC	84.44 ± 2.86	53.79 ± 2.28	63.91 ± 1.51	16.76 ± 2.50	66.00 ± 1.26	48.53 ± 2.35
KGC	81.90 ± 2.74	**61.58 ± 2.49**	63.45 ± 0.83	19.84 ± 2.23	67.80 ± 1.65	52.52 ± 2.81
KGC(A)	**86.23 ± 2.69**	57.25 ± 2.16	66.43 ± 0.92	17.17 ± 2.91	69.20 ± 1.37	**52.57 ± 2.22**
EIG	85.61 ± 1.69	56.08 ± 2.28	64.35 ± 1.43	12.19 ± 2.79	68.70 ± 1.71	49.57 ± 1.95
FULL	84.59 ± 2.51	54.37 ± 2.12	**67.51 ± 0.82**	**23.58 ± 2.50**	**69.90 ± 1.40**	52.57 ± 3.36

Table: Graph regression error on AQSOL (left) and ZINC (right). Method: GCN. $c = 0.3$

Methods	Test MAE	Train MAE	Epochs
VNGC	1.403 ± 0.005	0.629 ± 0.018	135.75
VEGC	1.390 ± 0.005	0.702 ± 0.003	107.75
MGC	1.447 ± 0.005	0.628 ± 0.012	111.00
SGC	1.489 ± 0.010	0.676 ± 0.021	107.00
KGC	1.389 ± 0.015	0.678 ± 0.013	112.00
KGC(A)	1.383 ± 0.005	0.657 ± 0.013	124.75
FULL	**1.372 ± 0.020**	0.593 ± 0.030	119.50

Methods	Test MAE	Train MAE	Epochs
VNGC	0.709 ± 0.005	0.432 ± 0.012	120.00
VEGC	0.646 ± 0.001	0.418 ± 0.008	138.25
MGC	0.677 ± 0.002	0.414 ± 0.006	112.50
SGC	0.649 ± 0.007	0.429 ± 0.008	111.75
KGC	0.737 ± 0.010	0.495 ± 0.012	113.50
KGC(A)	0.641 ± 0.003	0.433 ± 0.013	126.50
FULL	**0.416 ± 0.006**	0.313 ± 0.011	159.50

Interesting research topics

Can we define a spectrum-preserving restriction operator ($\mathbf{C}_w$) and prolongation operator for AMG?

Can we extend graph coarsening from graph classification/regression to node classification/regression?

Can we learn better graph coarsening (i.e., learn a dataset-dependent coarsening strategy)?

Can we use graph coarsening to produce a dictionary (i.e., motifs) for a collection of graphs?

Further Reading

(This talk) Chen et al. A Gromov–Wasserstein Geometric View of Spectrum-Preserving Graph Coarsening. *ICML*, 2023.

Note: Change of notation from paper to this talk:

- Partitioning: from $\{\mathcal{P}_1, \ldots, \mathcal{P}_n\}$ to $\{\mathcal{V}_1, \ldots, \mathcal{V}_n\}$
- GW distance: from $\langle \mathbf{M}, \mathbf{T} \rangle$ to $\langle \mathbf{C}, \mathbf{T} \rangle$
- Diagonal mass matrix: from $\mathbf{W}$ to $\mathbf{M}$

(Survey) Chen et al. Graph Coarsening: From Scientific Computing to Machine Learning. *SeMA Journal*, 2022.

(Learnable coarsening) Ma and Chen. Unsupervised Learning of Graph Hierarchical Abstractions with Differentiable Coarsening and Optimal Transport. *AAAI*, 2021.

(Algebraic distance for coarsening) Chen and Safro. Algebraic Distance on Graphs. *SISC*, 2011.

(Broader context of kernel K-means) Kokiopoulou et al. Trace Optimization and Eigenproblems in Dimension Reduction Methods. *NLAA*, 2011.

Check out papers at my homepage `https://jiechenjiechen.github.io`