Appendix A.

In the Appendix, we firstly provide the empirical summaries on the rank degeneracy phenomenon via GCN, GAT and HRGCN. Then we show the formal proofs of the previous statements and then show the details of the three experiments mentioned in the paper.

A.1. Rank degeneracy phenomenon

In this section, we first provide evidence for the rank degeneracy phenomenon for citation networks. Table 6 summarizes the rank of (re-weighted) adjacency matrices from GCN, GAT and HRGCN

Table 6: The rank of (re-weight) adjacency matrices of GCN, GAT and HRGCN

Datasets	**Cora**	**Citeseer**	**Pubmed**	**Computers**	**CS**	**Physics**	**Photo**
Number of Nodes	2708	3327	19717	13752	18333	34493	7650
Number of Repeated Rows	83	252	2902	35	115	81	15
Rank of $\hat{A}$	2401	2780	7596	13241	17146	33799	7501
Rank of $\theta \odot \hat{A}$	2638	3090	19604	13440	17817	33994	7473
Rank of $\eta \odot \hat{A}$	2708	3326	19699	13752	18330	34388	7641

From Table 6 one can check that rank degeneracy phenomenon widely exists in all commonly analyzed benchmarks. In particular, even the repeated rows are removed, the adjacency matrix ($\hat{A}$) utilized in GCN is still with large number of rank degeneracy whereas the re-weighted adjacency matrices in GAT ($\theta \odot \hat{A}$) and HRGCN ($\eta \odot \hat{A}$) are with much larger number of ranks. Furthermore, the rank of adjacency matrix ($\eta \odot \hat{A}$) in HRGCN has little difference to the number of nodes (which is the maximum possible rank of adjacency matrix) of the dataset, this indicates that our HRGCN not only can utilize the adjacency information from the matrix with repeat rows deleted, but also capable of distinguishing the nodes with the same connectivities. This shows the effectiveness of applying refined Ricci curvature in our model.

A.2. Formal proofs

Here we show the proof of Lemma 2, That is, in real practice, the probability of randomly simulating a rank degenerated attention matrix within the space that contains all possible attention matrix is 0.

Lemma 2 *Let* $S_1 := \{M \in \mathbb{R}^{n\times n} | m_{i,j} \geq 0, m_{i,j} = m_{j,i}, \sum_j m_{i,j} = 1 \forall i\}$ *be the space that contains all normalized matrices of size* $n \times n$*, with symmetric and positive entries. And* $S_2 \subset S_1, s.t. \forall M \in S_2, det(M) = 0$ *be the subset of all matrices with rank degeneracy from* S_1*. Let* μ *be a measure defined on* S_1*, then we have* $\mu(S_2) = 0$*.*

Proof It is easy to verify that S_1 defines a manifold $\mathcal{M}_1$ (multinomial symmetric and stochastic manifold), since all matrices contained in S_1 are symmetric and the summation of each row equals to 1, thus there are maximally $\frac{n(n-1)}{2}$ free elements in the matrices in

S_1. Hence we have the intrinsic dimension of $\mathcal{M}_1$ equal to $\frac{n(n-1)}{2}$.Similarly, S_2 defines a submanifold $\mathcal{M}_2 \subseteq \mathcal{M}_1$ with its dimension less than $\mathcal{M}_1$, this is because with one extra requirement ($\det(M) = 0$) introduced to all matrices in S_2, the degree of freedom of the matrices in S_1 will be at least decreased by 1. Let μ be a measure defined on S_1, due to the dimensionality difference, we have all matrices that belong to S_2 as measure 0 and that completes the proof. ■

Based on the claim on Lemma 2, we now verify our statement in Theorem 1. To show the HRGCN can balance the advantage within graph attention models in terms of expressive power.

Theorem 2 *Let D^*_{ATT} and D^*_{HRGCN} be the rank of $\theta \odot \hat{A}$ and $\eta \odot \hat{A}$, respectively, where θ is the matrix contains all learnable attention coefficients and η is the matrix with entries of the refined graph Ollivier Ricci curvature similarities that is:*

$$\eta_{ij} = Exp(-\widetilde{\kappa}_{ij})$$

Then we have $\mathcal{R}_{HRGCN} = \mathcal{R}_{ATT}$.

Proof Based on Lemma 2, let S_2 be the set that contains all possible matrices of $\theta \odot \hat{A}$, and we have S_2 is of full rank. Now, define $S_3 := \{M \in \mathbb{R}^{n\times n} | m_{i,j} \geq 0, m_{i,j} = m_{j,i}, \mathrm{Diag}(M) = 1 \forall i\}$ be the set that contain all possible matrices of $\eta \odot \hat{A}$. The diagonal entries of the matrices contained in S_3 are fixed as 1 since based on the definition of the refined Ricci curvature defined in equation (2) when $x_i = x_j$, we have their distance $d(x_i, x_j) = 0$ and this yields $m_{i,i} = \mathrm{Exp}(0) = 1$. Furthermore, compared to the matrices in S_1 defined in Lemma 2, matrices in S_3 do not have the row summation property ($\sum_j m_{i,j} = 1 \forall i$). Therefore, each row of the matrices in S_3 still only lost one degree of freedom due to the fixed diagonal values. Based on Lemma 2, one can define another measure μ_2 on S_1, then we have $\mu(S_3) = 0$. Hence matrices contained in S_3 are of full rank. Then based on Lemma 1 we have $\mathcal{R}_{HRGCN} = \mathcal{R}_{ATT}$, and that completes the proof. ■

We now prove the Lemma 3 included in the paper. Since Lemma 3 aims to show the propose HRGCN is potentially capable of handling the bottleneck problem mentioned in Topping et al. (2021) by preventing the long range dependency (negative curvatures) from being diluted in the original GCN. To prove lemma 3, we need the following proposition from Topping et al. (2021):

Proposition 1 (Theorem 2 in Topping et al. (2021)) *Given an unweighted graph $\mathcal{G}$, for any edge $e_{i,j} \in E$, we have $\kappa(i,j) \geq Ric(i,j)$.*

Here $\kappa(i,j)$ is the Ollivier Ricci curvature and $\mathrm{Ric}(i,j)$ is the balanced Forman curvature defined in Topping et al. (2021). Please refer to Topping et al. (2021) for the details of the proof of this proposition. We note that since we have $\kappa(i,j) \geq \mathrm{Ric}(i,j)$ and the negative balanced Forman curvature has been proved to be responsible for the over-squashing problem, hence when we have $\kappa_{i,j} < 0$ we must have $\mathrm{Ric} < 0$ and this illustrates that the negative $\kappa_{i,j}$ is also responsible to the over-squashing problem. With this conclusion in mind, we now provide the proof for Lemma 3.

Lemma 3 *Consider the propagation $H^{(\ell+1)} = \sigma_\ell(\mathcal{A}H^{(\ell)}W_\ell)$ at layer ℓ with $H^{(0)} = X$ and $\mathcal{A} = \Lambda \odot \hat{A}$ for some $\Lambda \in \mathbb{R}_+^{n\times n}$ as matrices with size $n \times n$ and all positive entry. Let $h_v^{(\ell)}$ represents the feature of node v at layer ℓ. Suppose $|\sigma'_\ell| \leq \alpha$ and $\|W_\ell\|_2 \leq \beta$ for all ℓ. Then we have for any node u, v with $d_{\mathcal{G}}(u, v) = \ell + 1$, we have $\left\|\frac{\partial h_v^{(\ell+1)}}{\partial x_u}\right\|_2 \leq (\alpha\beta)^{\ell+1}(\mathcal{A}^{\ell+1})_{uv}$.*

Proof First we see $h_v^{(\ell+1)} = \sigma_\ell(W_\ell^T(H^{(\ell)})^T a_v) = \sigma_\ell(\sum_{i=1}^n a_{vi}W_\ell^T h_i^{(\ell)})$, where we let $a_i^\top$ be the i-th row of matrix $\mathcal{A}$ and a_{ij} be the i, j-th entry of $\mathcal{A}$. Then by chain rule, we obtain

$$\begin{aligned}
&\left\|\frac{\partial h_v^{(\ell+1)}}{\partial x_u}\right\|_2 = \left\|\mathrm{diag}\Big(\sigma'_\ell(W_\ell^T(H^{(\ell)})^T a_v)\Big)\odot\Big(\sum_{i_\ell=1}^n a_{vi_\ell}W_\ell^T\frac{\partial h_{i_\ell}^{(\ell)}}{\partial x_u}\Big)\right\|_2\\
&\leq \alpha\left\|\sum_{i_\ell=1}^n a_{vi_\ell}W_\ell^T\frac{\partial h_{i_\ell}^{(\ell)}}{\partial x_u}\right\|_2\\
&\leq \alpha^{(\ell+1)}\left\|\sum_{i_\ell,i_{\ell-1},\dots,i_0} a_{vi_\ell}a_{i_\ell i_{\ell-1}}\cdots a_{i_1 i_0}W_\ell^T W_{\ell-1}^T\cdots W_0^T\frac{\partial h_{i_0}^{(0)}}{\partial x_u}\right\|_2\\
&= \alpha^{(\ell+1)}\Big(\sum_{i_\ell,i_{\ell-1},\dots,i_1} a_{vi_\ell}a_{i_\ell i_{\ell-1}}\cdots a_{i_1 u}\Big)\|W_\ell^T W_{\ell-1}^T\cdots W_0^T\|_2\\
&\leq (\alpha\beta)^{(\ell+1)}(\mathcal{A}^{\ell+1})_{uv}
\end{aligned}$$

where we apply the second inequality recursively to obtain the third inequality. ■

Lemma 3 shows when the derivative of activation functions and the weights are bounded, the sensitivity of the features on the input depend critically on the matrix $\mathcal{A}$, and this lead us to present the numerical verification (i.e., Table 2) in the main page.

A.2.1. Random Perturbation and Over-Smoothing

In this section, we show the reduction from tiny values of $\epsilon \sim U(0, 0.01)/1000$ s.t.$\epsilon <$ $\min(\eta_{i,j} \odot \hat{a}_{i,j})$ to the non-zero entries of $\eta \odot \hat{A}$ can help HRGCN to enjoy a higher distinguishability (expressive power) than GCN and attention based models, in the meanwhile, let HRGCN have a lower risk of over-smoothing than GCN. We show the advantages from ϵ in the next proposition.

Proposition 2 *Let $\mathcal{R}_{\mathcal{F},\theta}^{\epsilon}(HRGCN)$ and $\mathcal{R}_{\mathcal{F},\theta}(ATT)$ be the number of linear regions induced from HRGCN and attention based model, respectively. For any fixed input and output feature dimension as d_0 and d_1, we have:*

$$\mathcal{R}_{\mathcal{F},\theta}^{\epsilon}(HRGCN) > \mathcal{R}_{\mathcal{F},\theta}(ATT)$$

Proof Based on the proof of theorem 1 we have $\mathcal{R}_{\mathcal{F},\theta}(HRGCN) = \mathcal{R}_{\mathcal{F},\theta}(ATT)$, and the only situation that causes both HRGCN and attention based model lost their distinguishability is a graph or a subset of a graph that is complete and all nodes are with the same features. The tiny perturbation from ϵ to the non-zero entries of $\eta \odot \hat{A}$ addresses this issue by introducing the differences into the re-weighted matrix while preserving the connectivity of such complete graph (subset),thus we have $\mathcal{R}^{\epsilon}_{\mathcal{F},\theta}(HRGCN) > \mathcal{R}_{\mathcal{F},\theta}(ATT)$. ■

Now we show that the introduction of ϵ can potentially prevent HRGCN from the over-smoothing problem in the original GCN and GAT model Cai and Wang (2020). To show this, we firstly quantify the over-smoothing issue by defining graph Dirichlet energy as follows:

Definition 3 (Graph Dirichlet Energy) *Given node embedding matrix $X^{(l)} =: \{x_1^{(l)}, x_2^{(l)} ... x_N^{(l)}\}^T \in \mathbb{R}^{n \times d_l}$ learned from GCN at l-th layer, the Dirichlet energy $E(X^{(l)})$ is defined as:*

$$\begin{aligned} E(X^{(l)}) =& Tr(X^{(l)^T} \widetilde{\Delta} X^{(l)}) \\ =& \frac{1}{2} \sum_{i,j} w_{i,j} \left\| \frac{x_i^{(l)}}{\sqrt{1+d_i}} - \frac{x_j^{(l)}}{\sqrt{1+d_j}} \right\|_2^2, \end{aligned}$$

where $\widetilde{\Delta} = I - \hat{A}$ is the normalized graph Laplacian and $\hat{A}$ is the normalized adjacency matrix. The graph Dirichlet energy shows how smooth the information propagates in terms of GNN computation, and it has been reckoned as one of the metric that measures the over-smoothing issue in both GCN and GAT Cai and Wang (2020). Specifically, recall the computation within GCN can be described as:

$$H^{(\ell+1)} = \sigma(\hat{A} H^{(\ell)} W^{(\ell)}), \quad H^{(0)} = X,$$

If one were to remove the activation function σ, we have $\lim_{l \to \infty} \hat{A}^l H^{(0)} = H^{(\infty)}$, where each row of $H^{(\infty)}$ only depends on the degree of the corresponding node, meaning that the graph node features produced from the prior layer is irreducible and aperiodic. Thus the learning model loses discriminative information provided by the node features as the number of layers increases. Thanks to the next theorem we can show that by reducing ϵ to the product of $\eta \odot \hat{A}$, HRGCN produces a higher Dirichlet energy than GCN within any finite layers.

Theorem 3 *Let $\widetilde{A} = \eta \odot \hat{A}$ and $\widetilde{A}_\epsilon = \eta \odot \hat{A} - \epsilon$ be the re-weighted matrices of the curvature matrix η and the perturbed curvature matrix η_ϵ, respectively. Let $\widetilde{\Delta}$ and $\widetilde{\Delta}_\epsilon$ be the Laplacian matrices induced from $\widetilde{A}$ and $\widetilde{A}_\epsilon$, respectively. Then for any $\epsilon > 0$ and $\epsilon < min(\eta_{i,j} \odot \hat{a}_{i,j})$, at any specific layer (i.e., l-th layer), we have:*

$$E_\eta(X^{(l)}) < E_\eta(X^{(l)})_\epsilon,$$

where $E_\eta(X^{(l)})$ and $E_\eta(X^{(l)})_\epsilon$ are the Dirichlet energy at layer k induced from η with and without perturbation ϵ.

Proof The result can be easily proved by verification since we have:

$$\widetilde{A} = \eta \odot \hat{A} \ \text{ and } \ \widetilde{A}_\epsilon = \eta \odot \hat{A} - \epsilon$$

Table 7: Performance comparison between graph property prediction models. **QM7** is a regression task evaluated by MSE; **ogbn-molhiv** task by AUC-ROC percentage; others datasets are for classification and evaluated by test percentage accuracy. The values after ± are standard deviations. The top results are highlighted in **bold**.

Datasets	**PROTEINS**	**Mutagenicity**	**D&D**	**NCI1**	**ogb-molhiv**	**QM7**
TOLPooL	73.48±3.57	79.84±2.46	74.87±4.12	75.11±3.45	78.14±0.62	175.41±3.16
ATTENTION	73.93±5.37	80.25±2.22	77.48±2.65	74.04±1.27	74.44±2.12	177.99±2.22
SAGPooL	75.89±2.91	79.86±2.36	74.96±3.60	76.30±1.53	75.26±2.29	41.93±1.14
SUM	74.91±4.08	80.69±3.26	**78.91 ±3.37**	76.96±1.70	77.41±1.16	42.09±0.91
MAX	73.57±3.94	78.83±1.70	75.80±4.11	75.96±1.82	78.16±1.33	177.48±4.70
MEAN	73.13±3.18	80.37±2.44	76.89±2.23	73.70±2.55	78.21±0.90	177.49±4.69
HR_PooL	**76.77±2.15**	**81.49±3.15**	77.50±2.21	**77.1±3.25**	**79.40±2.51**	**150.24±3.25**

Then we have:

$$\widetilde{\Delta} = I - \eta \odot \hat{A} \text{ and } \widetilde{\Delta}_\epsilon = I - \eta \odot \hat{A} + \epsilon$$

For the perturbed graph Laplacian $\widetilde{\Delta}_\epsilon$ we have:

$$\begin{aligned} E(X^{(l)})_\epsilon &= \mathrm{Tr}({X^{(l)}}^T \widetilde{\Delta}_\epsilon X^{(l)}) \\ &= \mathrm{Tr}({X^{(l)}}^T (I - \eta \odot \hat{A} + \epsilon) X^{(l)} \\ &= \mathrm{Tr}({X^{(l)}}^T (\widetilde{\Delta} + \epsilon) X^{(l)}) \\ &= \mathrm{Tr}({X^{(l)}}^T \widetilde{\Delta} X^{(l)}) + \mathrm{Tr}({X^{(l)}}^T \epsilon X^{(l)}) \\ &= \mathrm{Tr}({X^{(l)}}^T \epsilon X^{(k)}) + E(X^{(l)}) \end{aligned}$$

Since $\epsilon > 0$ thus we have $\mathrm{Tr}({X^{(l)}}^T \epsilon X^{(l)}) > 0$ and therefore we have an positive increase of Dirichlet energy from ϵ ■

Hence we have proved that with the help of the random perturbation that initially assigned to HRGCN to increase its expressive power, we also lift system's Dirichlet energy to make HRGCN robust to over-smoothing.

A.3. Experiment Extend

The code for this paper can be found at
https://anonymous.4open.science/r/HRGCN-high-rank-GCN--ACCEPT/.

A.3.1. Curvature Assisted Graph Pooling

In this section, we show numerical results on (refined) curvature assisted graph pooling. Specifically, recall the self-attention graph pooling model Lee et al. (2019) in which the attention score is generated as: $Z = \sigma(\hat{A}X\theta_{att})$, where $\hat{A}$ is the normalized adjacency matrix and $\theta_{att} \in \mathbb{R}^{d_0 \times 1}$ is the attention coefficient matrix learned by the model. Since we have shown that HRGCN can produce the identical expressive power compared to attention based models, thus the refined Ricci curvature can naturally enhance the graph pooling schemes by illustrating the graph topological importance in terms of information (pooling) aggregation. Therefore, the curvature-based graph pooling model can be formulated as: $Z = \sigma(\eta \odot \hat{A}X)$, where $\eta_{i,j} = \text{Exp}(-\widetilde{\kappa}_{i,j})$, similar to attention pooling model, the pooling ratio $k \in (0, 1]$ is a hyperparameter that determines the number of nodes to keep. The top $[kn]$ nodes are selected based on the value of Z. Finally we equip the curvature pooling strategy into the HRGCN model and therefore the final attention score for HR_PooL can be expressed as: $Z = \sigma(\text{HRGCN}(X, \hat{A}))$.

Dataset Six benchmarks were selected to test the prediction power of HR_PooL , including four classification tasks with moderate sample size, one large scale classification task and one regression task. The classification tasks use the **TUDataset benchmarks** Morris et al. (2020) including **D&D** Dobson and Doig (2003), **PROTEINS** Borgwardt et al. (2005) to categorize proteins into enzyme and non-enzyme structures; **NCI1** Wale et al. (2008) to identify chemical compounds that block lung cancer cells; **Mutagenicity** Kazius et al. (2005) to recognize mutagenic molecular compounds for potentially marketable drug; and **QM7** Blum and Reymond (2009) to predict atomization energy value of molecules. The rest, namely **ogbn-molhiv** Hu et al. (2020) is used for large scale molecule classification.

Setup All the baseline models are with two fixed convolutional layers followed by one pooling layer as the network architecture. The graph convolution for the five **TUDatasets** uses the GCN model, and for **ogbg-molhiv** uses GIN with virtual nodes Ishiguro et al. (2019). Given graph representations, the prediction is made by a two-layer MLP, in which the hidden unit is identical to that of the convolutional layer. The parameters, including learning rate, weight decay, number of hidden units in the convolutional layer and drop out ratio, are fine-tunded using grid search mentioned earlier. The dataset was also split using standard data splitting method as the benchmark models did. Similar to the method mentioned in Zheng et al. (2021), the training stops when the validation loss stops improving for 20 consecutive epochs or reaching maximum 200 epochs. The accuracy results are averaged over 10 repetitions. For **TUDataset**, the mean test accuracy is reported, and for **ogbg-molhiv**, ROC-AUC score is used. The regression task on QM7 is reported as mean square error (MSE).

Baseline The learning outcome of RC-Pooling models are compared to seven baseline methods. These baselines are TOPKPooL Gao and Ji (2019), ATTTENTIONPooL Li et al. (2020), SAGPooL Lee et al. (2019), Zheng et al. (2021), and the classic SUM, MAX and MEAN pooling.

Results From Table 3 we can see the the propose pooling model in this paper outperforms the attention based pooling model in terms of both graph-level regression and classification tasks.

Table 8: Summary statistics for homophily citation networks. Moreover, the computational time for curvatures in these networks are: 1.99s, 2.38s, 20.8s,$\dot{3}$9.5s, 312.8s, 620s and 820s.

Datasets	#Classes	#nodes	#Edges	#Features	#Training	#Edges/#Nodes
Cora	7	2708	5429	1433	140	2.0
Citeseer	6	3327	4372	3703	120	1.42
Pubmed	3	19717	44338	500	60	2.25
Coauthor CS	15	18333	100227	6805	300	5.47
Coauthor Physics	5	34493	495924	8415	100	14.37
Amazon Computer	10	13381	259159	767	200	19.37
Amazon Photo	8	7487	126530	745	150	16.90

Table 9: Summary Statistics of the datasets, $H(G)$ represent the level of homophily of overall benchmark datasets

Datasets	#Class	#Feature	#Node	#Edge	Training	Validation	Testing	H(G)
Chameleon	5	2325	2277	31371	60%	20%	20%	0.247
Squirrel	5	2089	5201	198353	60%	20%	20%	0.216
Film	5	932	7600	26659	60%	20%	20%	0.221
Wisconsin	5	251	499	1703	60%	20%	20%	0.150
Texas	5	1703	183	279	60%	20%	20%	0.097
Cornell	5	1703	183	277	60%	20%	20%	0.386

A.3.2. Summary Statistics of the Datasets

In this section, we show some statistics of the graph datasets mentioned in the paper, and provide the sensitivity analysis on the hyperparameter α which is the initial mass assigned onto each node of the graph. As the graph Ricci curvature illustrates the connectivity importance between nodes, when $\alpha \approx 1$, indicating most of the initial mass are assigned to the nodes itself, causing the Wasserstein distance approaching to the shortest distance if the graph is unweighted and thus $\kappa_{i,j} = 1 - \frac{W_{i,j}}{d_{i,j}} \approx 0$. On the other hand, when $\alpha \approx 0$, the connectivity importance based on the $W_{i,j}$ value gradually appears. In the next few tables we show the benchmark statistics on (homophily) citation networks, (heterophily) benchmarks and datasets for graph Pooling. Moreover, similar to Ye et al. (2019), the computational complexity and time of the refined Ricci curvature for citation networks are also included.In addition, we also provide the hyperparameter searching spaces for both node classification and graph pooling.

Table 10: Summary statistics for the Graph Pooling Benchmarks, the letter R in the class number of **QM7** represents a regression task

Datasets	**PROTEINS**	**Mutagenicity**	**D&D**	**NCI1**	**ogbg-molhiv**	**QM7**
#Graphs	1113	4337	1178	4110	41127	7165
Min #Nodes	4	4	30	3	2	4
Max # Nodes	620	417	5748	111	222	23
Avg#Nodes	39	30	284	30	26	15
Avg#Edges	73	31	716	32	28	123
#Features	3	14	89	37	9	0
#Classes	2	2	2	2	2	1(R)

Hyperparameter Tuning Space We tuned the hyperparameters with the following selection of values. For learning rate:$\{0.1, 0.05, 0.01, 0.005\}$, number of hidden units in $\{16, 32, 64, 96\}$, weight decay in $\{0.001, 0.005, 0.01, 0.05\}$ and scale in $\{0.1, 0.5, 0.7, 0.9\}$ for **Cora, Citeseer and Pubmed**,$\{7, 8, 9, 10\}$ for **CS, Physics, Computers** and **Photo**. For the homophily graph and the graph dataset used in pooling we fixed Ollivier $\alpha = 0.9$ whereas for heterophily graphs we fixed $\alpha = 0.4$. The following tables shows the summary statistics of the datasets experimented in the paper.

A.3.3. Computational Complexity for Graph Ricci Curvature

The exact computation of graph Ricci curvature for large graph is somehow time costly since a learning programming problem need to be solved on each edge of the graph. Based on Ye et al. (2019), on each edge, to obtain the Wasserstein distance between the distributions generated from the probability measure function, the learn programming is conducted with $d_x \times d_y$ variables and $d_x + d_y$ constraints. Using the interior point solver (ECOS), the complexity is $\mathcal{O}((d_x \times d_y)^w$ in which w is the exponent of the complexity of matrix multiplication (the best known is 2.373). However, there are many approximation methods that can relax the computation of optimal transportation such as Sinkhorn Algorithm Cuturi (2013) and some methods can increase the precision of the Wasserstein distance for example Shi et al. (2021) and has proved to have almost identical computational complexity to the classic OT algorithms. We included the computation time for citation networks in Table 8.