Transformers Generalize DeepSets and Can be Extended to Graphs and Hypergraphs

Jinwoo Kim, Saeyoon Oh, Seunghoon Hong

School of Computing, KAIST

{jinwoo-kim, saeyoon17, seunghoon.hong}@kaist.ac.kr

We present a generalization of Transformers to sets, graphs, and hypergraphs, and reduce its computational cost to linear to input size.

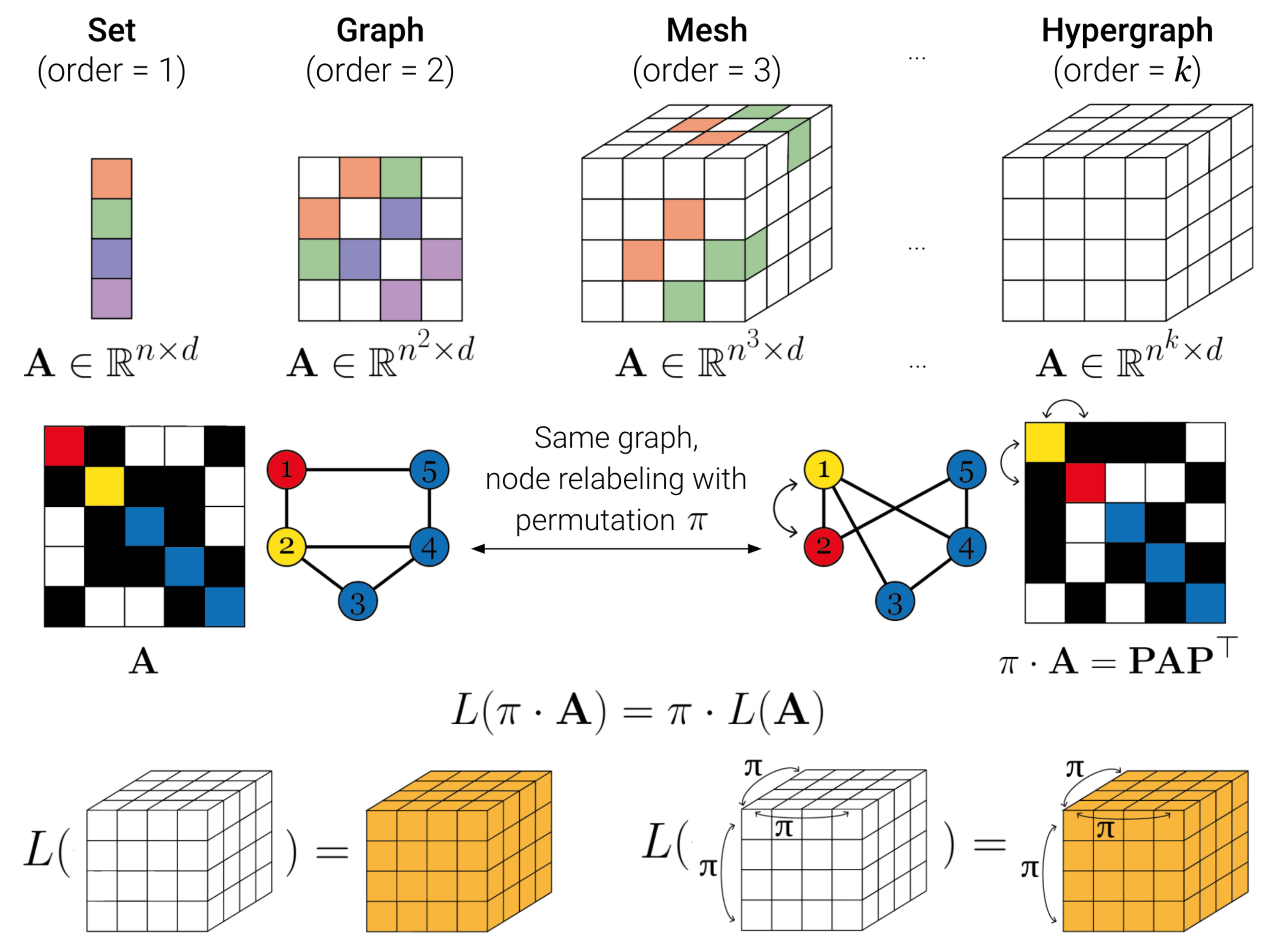
- Current graph neural nets are local message-passing (MPNNs), and do not scale well
- Equivariant MLPs are theoretically powerful and flexible, but less practical

Higher-Order Transformers offer a working solution

- Equivariance theory + self-attention → Transformers for any-order graphs
- Powerful operations, involving both local and global dependency over input elements
- Flexible translation between different-order graphs (e.g., set-to-(hyper)graph)
- Theoretically and empirically stronger than MPNNs, even with same linear complexity

Background: Permutation Equivariant Graph Learning

- View sets, graphs, and hypergraphs as permutable tensors; use equivariant layers that preserve isomorphism to process them



Background: Equivariant Linear Layers $L_{k\to l}$: $\mathbb{R}^{n^k \times d} \to \mathbb{R}^{n^l \times d'}$

- Theoretically maximally expressive [1], involving various local and global interactions

$$L_{k\to l}(\mathbf{A})_\mathbf{j} = \sum_\mu \sum_\mathbf{i} \mathbf{B}^\mu_{\mathbf{i},\mathbf{j}} \mathbf{A}_\mathbf{i} w_\mu + \sum_\lambda \mathbf{C}^\lambda_\mathbf{j} b_\lambda$$

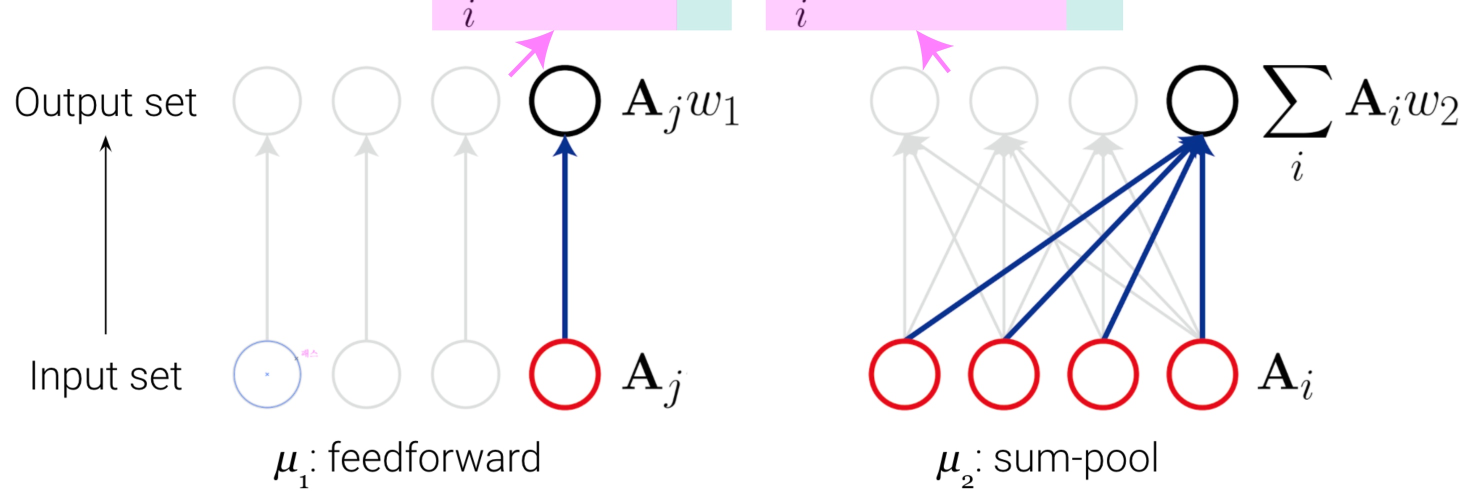
Outer sum over equivalence classes μ

Masked inner sum with binary basis tensor $\mathbf{B}^\mu$

$$\mathbf{B}^\mu_{\mathbf{i},\mathbf{j}} = \begin{cases} 1 & (\mathbf{i},\mathbf{j}) \in \mu \\ 0 & \text{otherwise} \end{cases}$$

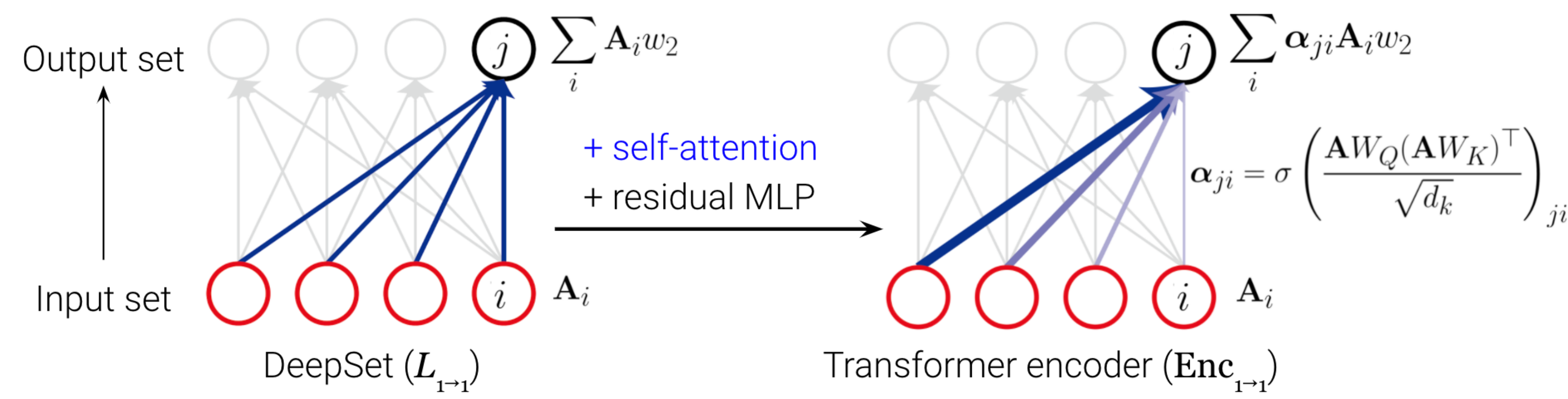
- Example: First-order equivariant layer $L_{1\to 1}$ (DeepSet)

$$L_{1\to 1}(\mathbf{A})_j = \sum_i (I_n)_{ij} \mathbf{A}_i w_1 + \sum_i (1_n 1_n^\top)_{ij} \mathbf{A}_i w_2 + (1_n)_j b_1$$



Transformers ($\mathrm{Enc}_{1\to 1}$) Generalize DeepSets ($L_{1\to 1}$)

- DeepSet, or first-order linear layer ($L_{1\to 1}$), is feedforward (μ_1) + *static* sum-pool (μ_2)
- To model *adaptive* interactions of set elements, we use self-attention mechanism
- This procedurally improves a DeepSet layer into a Transformer encoder layer ($\mathrm{Enc}_{1\to 1}$)



Higher-Order Transformers $\mathrm{Enc}_{k\to l}$

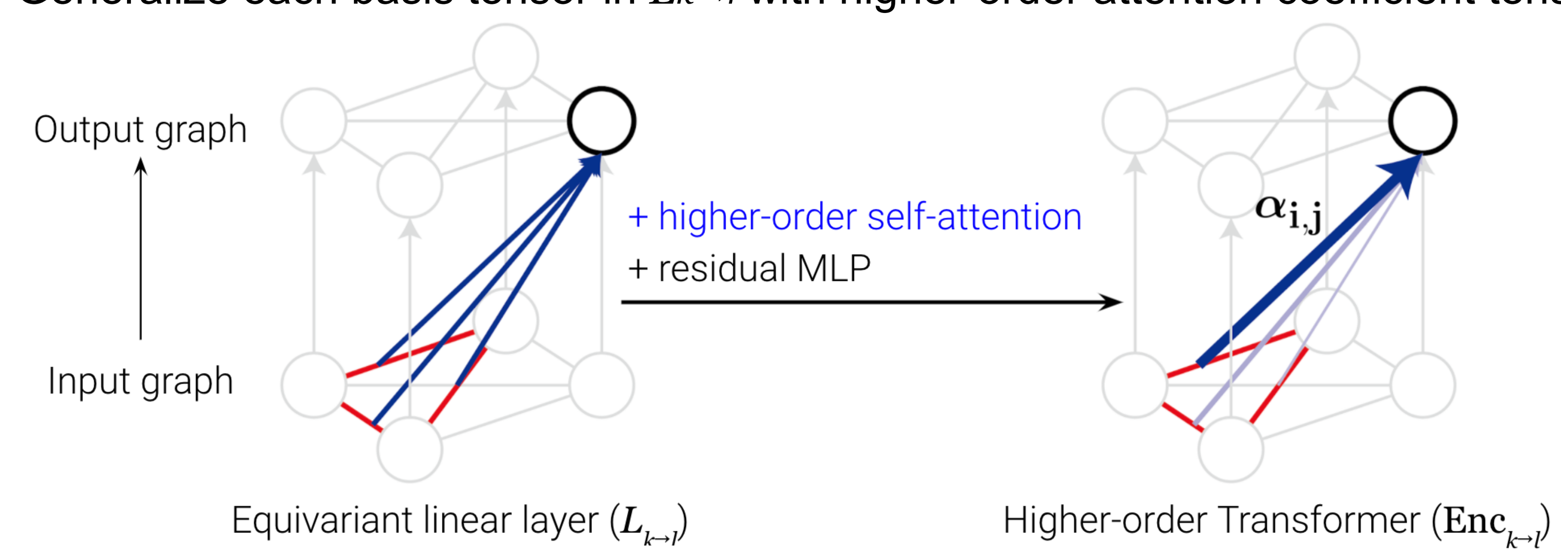
- Extend the first-order case (set) to higher orders (graphs and hypergraphs)
- Combine higher-order self-attention $\mathrm{Attn}_{k\to l}$ and residual equivariant $\mathrm{MLP}_{l\to l}$

$$\mathrm{Enc}_{k\to l}(\mathbf{A}) = \mathrm{Attn}_{k\to l}(\mathbf{A}) + \mathrm{MLP}_{l\to l}(\mathrm{Attn}_{k\to l}(\mathbf{A}))$$

$$\mathrm{MLP}_{l\to l}(\cdot) = L^2_{l\to l}(\mathrm{ReLU}(L^1_{l\to l}(\cdot)))$$

Higher-Order Self-Attention $\mathrm{Attn}_{k\to l}$

- Generalize each basis tensor in $L_{k\to l}$ with higher-order attention coefficient tensor



$$\mathrm{Attn}_{k\to l}(\mathbf{A})_\mathbf{j} = \sum_{h=1}^{H} \sum_\mu \sum_\mathbf{i} \boldsymbol{\alpha}^{h,\mu}_{\mathbf{i},\mathbf{j}} \mathbf{A}_\mathbf{i} w^V_{h,\mu} w^O_{h,\mu},$$

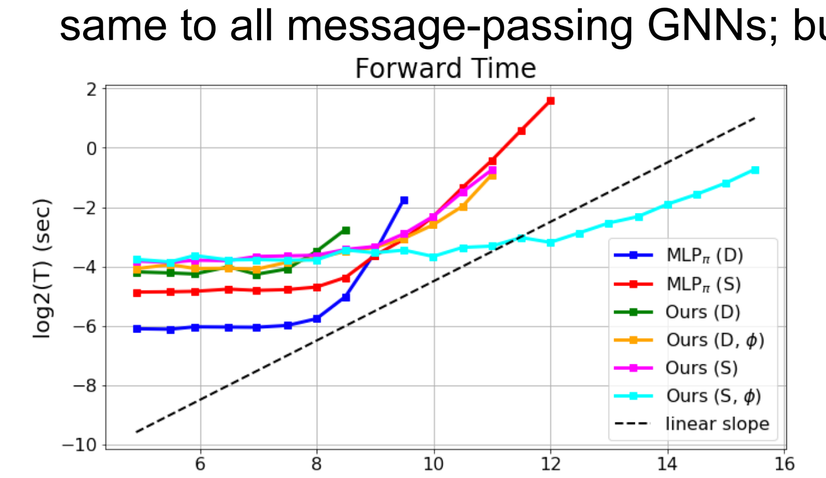
$$\boldsymbol{\alpha}^\mu_{\mathbf{i},\mathbf{j}} = \begin{cases} \sigma(\mathbf{Q}^\mu_\mathbf{j}, \mathbf{K}^\mu_\mathbf{i})/Z_\mathbf{j} & (\mathbf{i},\mathbf{j}) \in \mu \\ 0 & \text{otherwise} \end{cases} \quad \text{where} \quad \begin{aligned} \mathbf{Q}^\mu &= L^\mu_{k\to l}(\mathbf{A}) \\ \mathbf{K}^\mu &= L^\mu_{k\to k}(\mathbf{A}) \end{aligned}$$

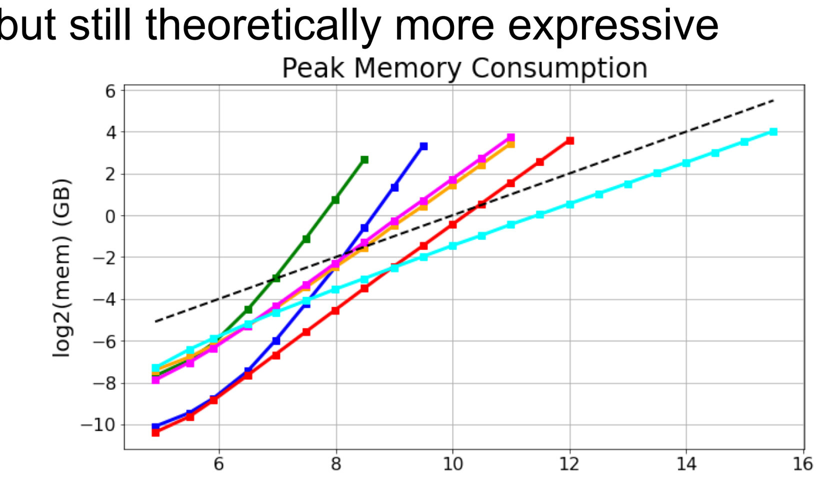
Higher-order self-attention coefficient

Tensorized query/key

Asymptotically Efficient Higher-Order Transformers $\mathrm{Enc}_{k\to l,\varphi}$

- Reduce asymptotic complexity of $\mathrm{Enc}_{k\to l}$
  + Lightweight sublayers
  + Sparse input and output hypergraphs
  + Kernelized attention
- Resulting architecture has linear complexity $O(m)$ to number of input hyperedges m, same to all message-passing GNNs; but still theoretically more expressive





Large-Scale Graph Regression (2→2, 2→0): PCQM4M-LSC

- Higher-order Transformer outperforms all baselines by a large margin, demonstrating benefits in large-scale settings
- Higher-order attention is potentially better in handling long-range interactions than the current practice of augmenting MPNNs with a virtual node
- Heuristic graph embeddings (e.g., Laplacian) are insufficient to utilize features from edges, while second-order Transformers can use all edge information

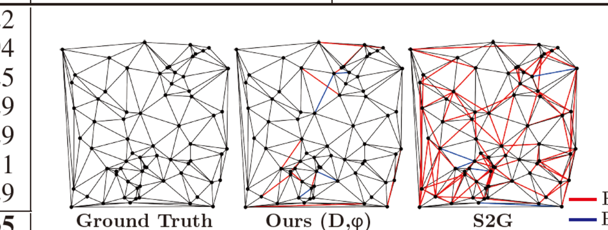
Model	Validate MAE
MLP-FINGERPRINT ([17])	0.2044
GCN ([17])	0.1684
GIN ([17])	0.1536
GCN-VN ([17])	0.1510
GIN-VN ([17])	0.1396
Transformer + Laplacian PE*	0.2162
MLP$_\pi$ (S)*	0.1464
Ours (S, ϕ)$_{\text{-SMALL}}$*	0.1376
Ours (S, ϕ)*	0.1294
Ours (S, ϕ)	**0.1263**

Set-to-Graph Prediction (1→2): Delaunay, Jets

- Mixed-order Transformers, both softmax and kernel, outperform all baselines; kernelized attention is often competitive or sometimes better than softmax
- Compared to equivariant MLP, the results indicate that attention mechanism is helpful in modeling graphs with varying numbers of nodes

	Method	F1	RI	ARI
Jets (B)	AVR	0.565	0.612	0.318
	MLP	0.533	0.643	0.315
	SIAM	0.606	0.675	0.411
	SIAM-3	0.597	0.673	0.396
	GNN	0.586	0.661	0.381
	S2G	0.646	0.736	0.491
	S2G+	0.655	0.747	0.508
	Ours (D)	0.667	0.746	0.520
	Ours (D, ϕ)	**0.670**	**0.751**	**0.526**
Jets (C)	AVR	0.695	0.650	0.326
	MLP	0.686	0.658	0.319
	SIAM	0.729	0.695	0.406
	SIAM-3	0.719	0.710	0.421
	GNN	0.720	0.689	0.390
	S2G	0.747	0.727	0.457
	S2G+	0.751	0.733	0.467
	Ours (D)	0.755	0.732	0.469
	Ours (D, ϕ)	**0.757**	**0.735**	**0.473**
Jets (L)	AVR	0.970	0.965	0.922
	MLP	0.960	0.957	0.894
	SIAM	0.973	0.970	0.925
	SIAM-3	0.895	0.876	0.729
	GNN	0.972	0.970	0.929
	S2G	0.972	0.970	0.931
	S2G+	0.971	0.969	0.929
	Ours (D)	**0.974**	**0.972**	**0.935**
	Ours (D, ϕ)	**0.974**	**0.972**	**0.935**

	Method	Acc	Prec	Rec	F1
Delaunay (50)	SIAM	0.939	0.766	0.653	0.704
	SIAM-3	0.911	0.608	0.538	0.570
	GNN0	0.826	0.384	0.966	0.549
	GNN5	0.809	0.363	**0.985**	0.530
	GNN10	0.759	0.311	0.978	0.471
	S2G	0.984	0.927	0.926	0.926
	S2G+	0.983	0.927	0.925	0.926
	Ours (D)	**0.994**	**0.981**	0.967	**0.974**
	Ours (D, ϕ)	0.991	0.967	0.952	0.959
Delaunay (20-80)	SIAM	0.919	0.667	0.764	0.687
	SIAM-3	0.895	0.578	0.622	0.587
	GNN0	0.810	0.387	0.946	0.536
	GNN5	0.777	0.352	**0.975**	0.506
	GNN10	0.746	0.322	0.970	0.474
	S2G	0.947	0.736	0.934	0.799
	S2G+	0.947	0.735	0.934	0.798
	Ours (D)	**0.993**	**0.982**	0.960	**0.971**
	Ours (D, ϕ)	0.989	0.948	0.956	0.952



k-Uniform Hyperedge Prediction (1→k): GPS, MovieLens, Drug

- Higher-order Transformer generally shows high performance, even without introducing task-specific inductive biases as in some baselines
- Higher-order self-attention is effective in learning higher-order representations

	GPS		MovieLens		Drug	
	AUC	AUPR	AUC	AUPR	AUC	AUPR
node2vec-mean ([36])	0.563	0.191	0.562	0.197	0.670	0.246
node2vec-min ([36])	0.570	0.185	0.539	0.186	0.684	0.258
DHNE ([36])	0.910	0.668	0.877	0.668	0.925	0.859
Hyper-SAGNN-E	0.947	0.788	0.922	**0.792**	0.963	0.897
Hyper-SAGNN-W	0.907	0.632	0.909	0.683	0.956	0.890
S2G+ (S)	0.943	0.726	0.918	0.737	0.963	0.898
Ours (S, ϕ)	**0.952**	**0.804**	**0.923**	0.771	**0.964**	**0.901**

1. Maron et al., Invariant and Equivariant Graph Networks, 2019.

KAIST

NEURAL INFORMATION PROCESSING SYSTEMS