Equivariant Subgraph Aggregation Networks

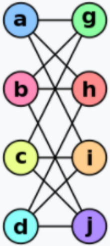
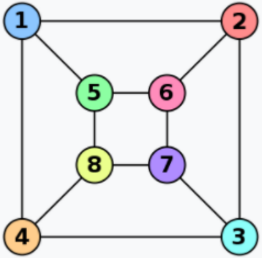
Beatrice Bevilacqua Fabrizio Frasca Derek Lim
Balasubramaniam Srinivasan Chen Cai Gopinath Balamurugan
Michael M. Bronstein Haggai Maron
presenter: Shen Yuan

中国人民大學
RENMIN UNIVERSITY OF CHINA
高瓴人工智能学院
Gaoling School of Artificial Intelligence

- Introduction
- Equivariant Subgraph Aggregation Networks(ESAN)
- A WL Analogue for ESAN
- Experiments
- Summary

Preliminary

Graph Isomorphism Two graphs are considered isomorphic if there is a mapping between the nodes of the graphs that preserves node adjacencies.

Graph G	Graph H	An isomorphism between G and H
		$f(a) = 1$ $f(b) = 6$ $f(c) = 8$ $f(d) = 3$ $f(g) = 5$ $f(h) = 2$ $f(i) = 4$ $f(j) = 7$

Preliminary

1-dimensional Weisfeiler-Leman (1-WL) test It’s a simple iterative algorithm to distinguish two graphs, which can produce for each graph a canonical form.

Preliminary

1-dimensional Weisfeiler-Leman (1-WL) test It's a simple iterative algorithm to distinguish two graphs, which can produce for each graph a canonical form.

- If the canonical forms of two graphs are **not equivalent**, then the graphs are definitively not isomorphic.

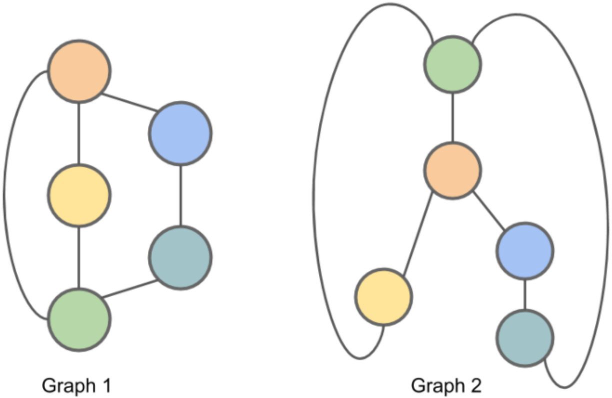
Preliminary

1-dimensional Weisfeiler-Leman (1-WL) test It's a simple iterative algorithm to distinguish two graphs, which can produce for each graph a canonical form.

- If the canonical forms of two graphs are **not equivalent**, then the graphs are definitively not isomorphic.
- If the canonical forms of two graphs are **equivalent**, the graphs may be isomorphic.

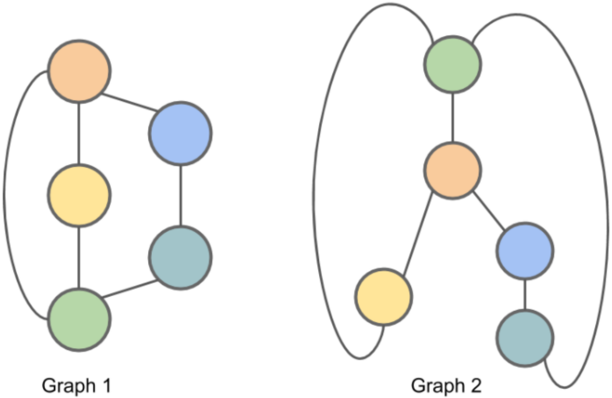
Test!

Are these two graphs isomorphic?

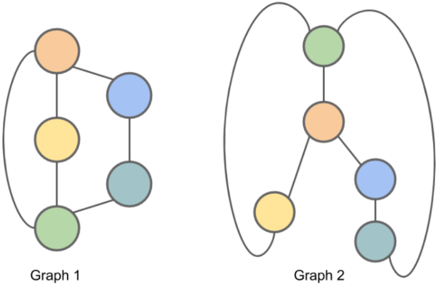


Test!

Are these two graphs isomorphic? YES!



1-dimensional Weisfeiler-Leman (1-WL) Test



$$c_v^{t+1} \leftarrow \mathrm{HASH}(c_v^t, N_v^t)$$

Motivation

While two graphs may not be distinguishable by 1-WL test, they often contain distinguishable subgraphs.

Motivation

While two graphs may not be distinguishable by 1-WL test, they often contain distinguishable subgraphs.

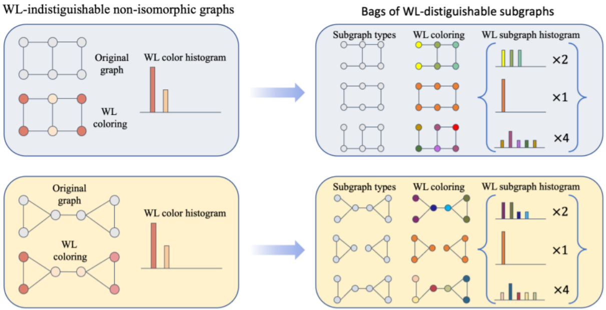


Figure 1: We present a provably expressive graph learning framework based on representing graphs as bags of subgraphs and processing them with an equivariant architecture, composed of GNNs and set networks. **Left panel:** A pair of graphs not distinguishable by the WL test. **Right panel:** The corresponding bags (multisets) of all edge-deleted subgraphs, which can be distinguished by our framework.

Contribution

- This paper proposed a framework called **Equivariant Subgraph Aggregation Networks(ESAN)** to improve expressive power of MPNNs.
- It developed variants **DS(S)-WL** of **the 1-dimensional Weisfeiler-Leman (1-WL) test** for graph isomorphism.

- Introduction
- Equivariant Subgraph Aggregation Networks(ESAN)
- A WL Analogue for ESAN
- Experiments
- Summary

Equivariant Subgraph Aggregation Networks(ESAN)

The ESAN framework consists of

- Neural network architectures for processing bags of subgraphs (**DSS-GNN** and **DS-GNN**)

Equivariant Subgraph Aggregation Networks(ESAN)

The ESAN framework consists of

- Neural network architectures for processing bags of subgraphs (**DSS-GNN** and **DS-GNN**)
- Subgraph selection policies

DSS-GNN

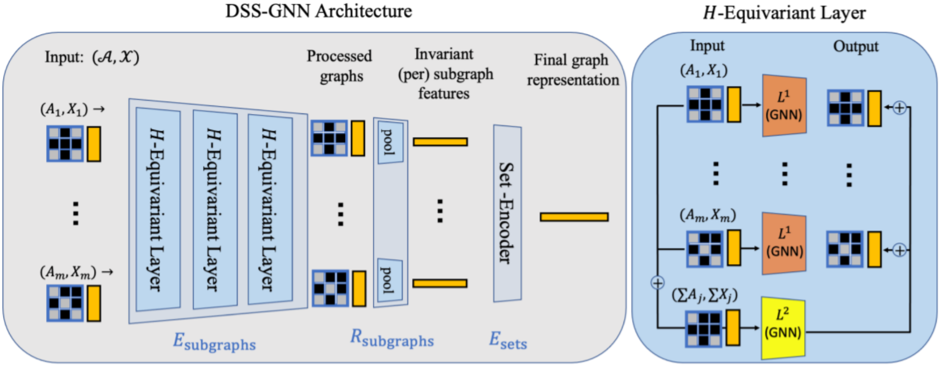


Figure 3: DSS-GNN layers and architecture. **Left panel**: the DSS- GNN architecture is composed of three blocks: a Feature Encoder, a Readout Layer and a Set Encoder. **Right panel**: a DSS-GNN layer is constructed from a Siamese part (orange) and an information-sharing part (yellow).

Bags-of-Graphs Encoder Architecture

- The bag(multiset) $S_G = \{G_1, \ldots, G_m\}$ of subgraphs of G can be represented as tensor $(\mathcal{A}, \mathcal{X}) \in \mathbb{R}^{n \times n \times m} \times \mathbb{R}^{n \times d \times m}$

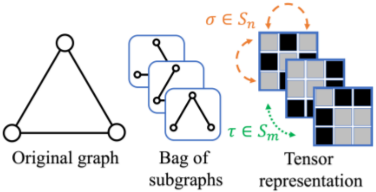


Figure 2: The symmetry structure of a bag of subgraphs, in this case the set of all $m = 3$ edge-deleted subgraphs. This set of subgraphs is represented as an $m \times n \times n$ tensor $\mathcal{A}$ (and additional node features that are not illustrated here). $(\tau, \sigma) \in S_m \times S_n$ acts on the tensor $\mathcal{A}$ by permuting the subgraphs (τ) and the nodes in the subgraphs (σ), which are assumed to be ordered consistently.

Bags-of-Graphs Encoder Architecture

- The bag(multiset) $S_G = \{G_1, \ldots, G_m\}$ of subgraphs of G can be represented as tensor $(\mathcal{A}, \mathcal{X}) \in \mathbb{R}^{n \times n \times m} \times \mathbb{R}^{n \times d \times m}$
- n denotes the number of nodes and m the number of subgraphs

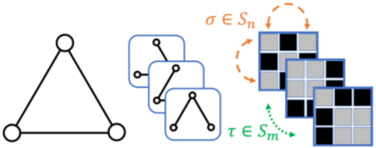


Figure 2: The symmetry structure of a bag of subgraphs, in this case the set of all $m = 3$ edge-deleted subgraphs. This set of subgraphs is represented as an $m \times n \times n$ tensor $\mathcal{A}$ (and additional node features that are not illustrated here). $(\tau, \sigma) \in S_m \times S_n$ acts on the tensor $\mathcal{A}$ by permuting the subgraphs (τ) and the nodes in the subgraphs (σ), which are assumed to be ordered consistently.

Bags-of-Graphs Encoder Architecture

- The bag(multiset) $S_G = \{G_1, \dots, G_m\}$ of subgraphs of G can be represented as tensor $(\mathcal{A}, \mathcal{X}) \in \mathbb{R}^{n\times n\times m} \times \mathbb{R}^{n\times d\times m}$
- n denotes the number of nodes and m the number of subgraphs
- $\mathcal{A} \in \mathbb{R}^{n\times n\times m}$ represents a set of m adjacency matrices, and $\mathcal{X} \in \mathbb{R}^{n\times d\times m}$ represents a set of m node feature matrices.

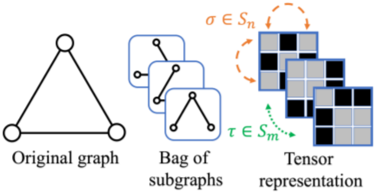


Figure 2: The symmetry structure of a bag of subgraphs, in this case the set of all $m = 3$ edge-deleted subgraphs. This set of subgraphs is represented as an $m \times n \times n$ tensor $\mathcal{A}$ (and additional node features that are not illustrated here). $(\tau, \sigma) \in S_m \times S_n$ acts on the tensor $\mathcal{A}$ by permuting the subgraphs (τ) and the nodes in the subgraphs (σ), which are assumed to be ordered consistently.

Bags-of-Graphs Encoder Architecture

- The bag(multiset) $S_G = \{G_1, \ldots, G_m\}$ of subgraphs of G can be represented as tensor $(\mathcal{A}, \mathcal{X}) \in \mathbb{R}^{n\times n\times m} \times \mathbb{R}^{n\times d\times m}$
- n denotes the number of nodes and m the number of subgraphs
- $\mathcal{A} \in \mathbb{R}^{n\times n\times m}$ represents a set of m adjacency matrices, and $\mathcal{X} \in \mathbb{R}^{n\times d\times m}$ represents a set of m node feature matrices.
- $\sigma \in S_n$ means node permutations, $(\sigma \cdot A)_{ij} = A_{\sigma^{-1}(i)\sigma^{-1}(j)}$, $(\sigma \cdot X)_{il} = X_{\sigma^{-1}(i)l}$ $\tau \in S_m$ means subgraph permutations, $(\tau \cdot \mathcal{A})_{ijk} = \mathcal{A}_{ij\tau^{-1}(k)}$, $(\tau \cdot \mathcal{X})_{ilk} = \mathcal{X}_{il\tau^{-1}(k)}$

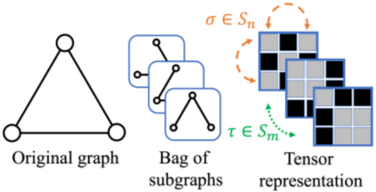
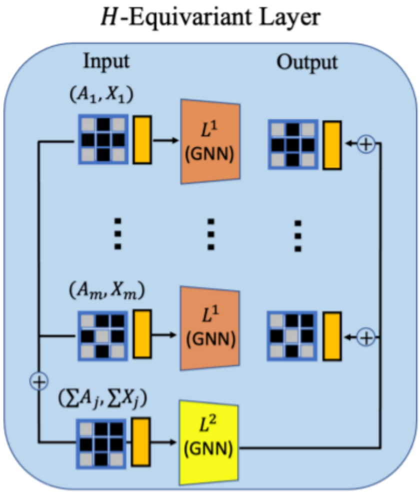


Figure 2: The symmetry structure of a bag of subgraphs, in this case the set of all $m = 3$ edge-deleted subgraphs. This set of subgraphs is represented as an $m \times n \times n$ tensor $\mathcal{A}$ (and additional node features that are not illustrated here). $(\tau, \sigma) \in S_m \times S_n$ acts on the tensor $\mathcal{A}$ by permuting the subgraphs (τ) and the nodes in the subgraphs (σ), which are assumed to be ordered consistently.

H-equivariant layers

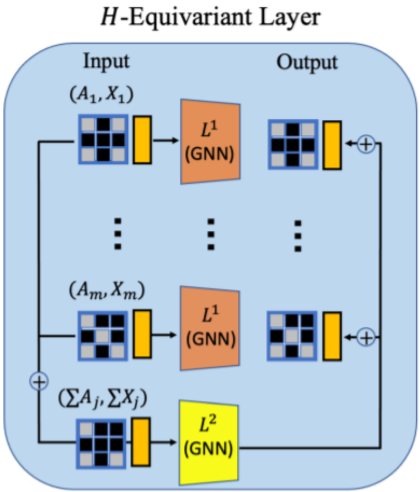
- $L : \mathbb{R}^{n\times n\times m} \times \mathbb{R}^{n\times d\times m} \to \mathbb{R}^{n\times n\times m} \times \mathbb{R}^{n\times d'\times m}$ map bags of subgraphs to bags of subgraphs:



H-equivariant layers

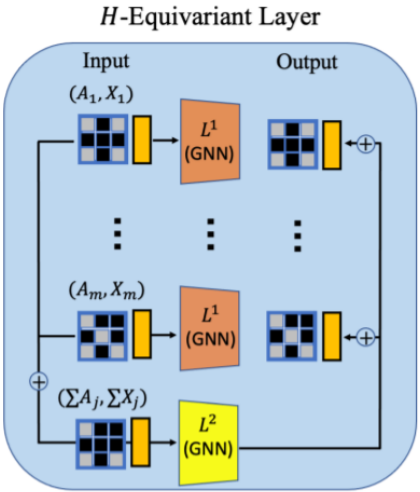
- $L : \mathbb{R}^{n\times n\times m} \times \mathbb{R}^{n\times d\times m} \to \mathbb{R}^{n\times n\times m} \times \mathbb{R}^{n\times d'\times m}$ map bags of subgraphs to bags of subgraphs:
-

$$(L(\mathcal{A}, \mathcal{X}))_i = L^1(\mathcal{A}_i, \mathcal{X}_i) + L^2(\sum_{j=1}^{m} A_j, \sum_{j=1}^{m} X_j)$$



H-equivariant layers

- $L : \mathbb{R}^{n\times n\times m} \times \mathbb{R}^{n\times d\times m} \to \mathbb{R}^{n\times n\times m} \times \mathbb{R}^{n\times d'\times m}$ map bags of subgraphs to bags of subgraphs:
- $$(L(\mathcal{A},\mathcal{X}))_i = L^1(\mathcal{A}_i,\mathcal{X}_i) + L^2\Big(\sum_{j=1}^{m} A_j, \sum_{j=1}^{m} X_j\Big)$$
- $L^1, L^2 : \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times d'}$ represent two graph encoders and can be any type of GNN layer.



Subgraph Selection Policies

This paper explores four simple subgraph selection policies:

- The **node-deleted policy(ND)**, a graph is mapped to the set containing all subgraphs that can be obtained from the original graph by removing a single node

Subgraph Selection Policies

This paper explores four simple subgraph selection policies:

- The **node-deleted policy(ND)**, a graph is mapped to the set containing all subgraphs that can be obtained from the original graph by removing a single node
- The **edge-deleted policy(ED)** is defined by removing a single edge

Subgraph Selection Policies

This paper explores four simple subgraph selection policies:

- The **node-deleted policy(ND)**, a graph is mapped to the set containing all subgraphs that can be obtained from the original graph by removing a single node
- The **edge-deleted policy(ED)** is defined by removing a single edge
- The **ego-networks policy(EGO)** maps each graph to a set of ego-networks of some specified depth, one for each node in the graph (a k-Ego-network of a node is its k-hop neighbourhood with the induced connectivity).

Subgraph Selection Policies

This paper explores four simple subgraph selection policies:

- The **node-deleted policy(ND)**, a graph is mapped to the set containing all subgraphs that can be obtained from the original graph by removing a single node
- The **edge-deleted policy(ED)** is defined by removing a single edge
- The **ego-networks policy(EGO)** maps each graph to a set of ego-networks of some specified depth, one for each node in the graph (a k-Ego-network of a node is its k-hop neighbourhood with the induced connectivity).
- The **EGO+** is a variant of the **EGO** where the root node holds an identifying feature

- Introduction
- Equivariant Subgraph Aggregation Networks(ESAN)
- A WL Analogue for ESAN
- Experiments
- Summary

DSS-WL and DS-WL

This paper proposed DSS-WL and DS-WL that are variants of 1-WL test. The only difference is **refinement** step.

$$c_{v,S}^{t+1} \leftarrow \mathrm{HASH}(c_{v,S}^{t}, N_{v,S}^{t}, C_{v}^{t}, M_{v}^{t})$$

- $N_{v,S}^t$ denotes the multiset of colors in v's neighborhood over subgraph S
- C_v^t represents the multiset of v's colors across subgraphs
- M_v^t denotes the multiset of colors in v's neighborhood over original graph G

DSS-WL and DS-WL

This paper proposed DSS-WL and DS-WL that are variants of 1-WL test. The only difference is **refinement** step.

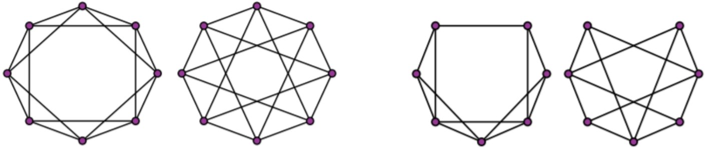
$$c_{v,S}^{t+1} \leftarrow \mathrm{HASH}(c_{v,S}^{t}, N_{v,S}^{t}, C_{v}^{t}, M_{v}^{t})$$

- $N_{v,S}^{t}$ denotes the multiset of colors in v's neighborhood over subgraph S
- C_{v}^{t} represents the multiset of v's colors across subgraphs
- M_{v}^{t} denotes the multiset of colors in v's neighborhood over original graph G

Is DS(S)-WL strictly more powerful than 1-WL?

Circulant Skip Link(CSL)

CSL$(n, 2)$ can be distinguished from any CSL(n, k) with $k \in [3, n/2 - 1]$ by DS-WL and DSS-WL with either the ND, EGO, or EGO+ policy.



(a) CSL$(8, 2)$ (left) and CSL$(8, 3)$ (right)

(b) Subgraphs from ND over CSL$(8, 2)$ and CSL$(8, 3)$

Figure 4: Graphs CSL$(8, 2)$ and CSL$(8, 3)$ (left) and their node-deleted subgraphs (right).

- Introduction
- Equivariant Subgraph Aggregation Networks(ESAN)
- A WL Analogue for ESAN
- Experiments
- Summary

Experiments

Table 1: TUDatasets. The top three are highlighted by **First**, **Second**, **Third**. Gray background indicates that ESAN outperforms the base encoder.

Dataset	MUTAG	PTC	PROTEINS	NCI1	NCI109	IMDB-B	IMDB-M
DCNN (Atwood & Towsley, 2016)	N/A	N/A	61.3±1.6	56.6±1.0	N/A	49.1±1.4	33.5±1.4
DGCNN (Zhang et al., 2018)	85.8±1.8	58.6±2.5	75.5±0.9	74.4±0.5	N/A	70.0±0.9	47.8±0.9
IGN (Maron et al., 2019b)	83.9±13.0	58.5±6.9	76.6±5.5	74.3±2.7	72.8±1.5	72.0±5.5	48.7±3.4
PPGNs (Maron et al., 2019a)	90.6±8.7	66.2±6.6	**77.2**±4.7	83.2±1.1	82.2±1.4	73.0±5.8	50.5±3.6
NATURAL GN (de Haan et al., 2020)	89.4±1.6	**66.8**±1.7	71.7±1.0	82.4±1.3	N/A	73.5±2.0	51.3±1.5
GSN (Bouritsas et al., 2020)	**92.2**±7.5	**68.2**±7.2	76.6±5.0	**83.5**±2.0	N/A	**77.8**±3.3	**54.3**±3.3
SIN (Bodnar et al., 2021b)	N/A	N/A	76.4±3.3	82.7±2.1	N/A	75.6±3.2	52.4±2.9
CIN (Bodnar et al., 2021a)	**92.7**±6.1	**68.2**±5.6	**77.0**±4.3	**83.6**±1.4	**84.0**±1.6	75.6±3.7	52.7±3.1
GIN (Xu et al., 2019)	89.4±5.6	64.6±7.0	76.2±2.8	82.7±1.7	82.2±1.6	75.1±5.1	52.3±2.8
GIN + ID-GNN (You et al., 2021)	90.4±5.4	67.2±4.3	75.4±2.7	82.6±1.6	82.1±1.5	76.0±2.7	52.7±4.2
DROPEDGE (Rong et al. (2019))	91.0±5.7	64.5±2.6	73.5±4.5	82.0±2.6	82.2±1.4	**76.5**± 3.3	52.8± 2.8
DS-GNN (GIN) (ED)	89.9±3.7	66.0±7.2	76.8±4.6	83.3±2.5	83.0±1.7	76.1±2.6	52.9±2.4
DS-GNN (GIN) (ND)	89.4±4.8	66.3±7.0	**77.1**±4.6	**83.8**±2.4	82.4±1.3	75.4±2.9	52.7±2.0
DS-GNN (GIN) (EGO)	89.9±6.5	68.6±5.8	76.7±5.8	81.4±0.7	79.5±1.0	76.1±2.8	52.6±2.8
DS-GNN (GIN) (EGO+)	91.0±4.8	68.7±7.0	76.7±4.4	82.0±1.4	80.3±0.9	**77.1**±2.6	53.2±2.8
DSS-GNN (GIN) (ED)	91.0±4.8	66.6±7.3	75.8±4.5	83.4±2.5	82.8±0.9	76.8±4.3	53.5±3.4
DSS-GNN (GIN) (ND)	91.0±3.5	66.3±5.9	76.1±3.4	83.6±1.5	**83.1**±0.8	76.1±2.9	53.3±1.9
DSS-GNN (GIN) (EGO)	91.0±4.7	68.2±5.8	76.7±4.1	83.6±1.8	82.5±1.6	76.5±2.8	53.3±3.1
DSS-GNN (GIN) (EGO+)	91.1±7.0	**69.2**±6.5	75.9±4.3	83.7±1.8	82.8±1.2	**77.1**±3.0	53.2±2.4
GRAPHCONV (Morris et al., 2019)	90.5±4.6	64.9±10.4	73.9±6.1	82.4±2.7	81.7±1.0	76.1±3.9	**53.1**±2.9
GRAPHCONV + ID-GNN (You et al., 2021)	89.4±4.1	65.4±7.1	71.9±4.6	83.4±2.4	**82.9**±1.2	76.1±2.5	**53.7**±3.3
RNI (Abboud et al., 2020)	91.0±4.9	64.3±6.1	73.3±3.3	82.1±1.7	81.7±1.0	75.5±3.3	**53.1**±1.9
DS-GNN (GRAPHCONV) (ED)	90.4±4.1	65.7±5.2	76.3±5.2	82.7±1.9	82.4±1.5	75.3±2.3	53.5±2.3
DS-GNN (GRAPHCONV) (ND)	88.3±5.1	66.6±7.8	76.8±3.9	82.9±2.5	82.7±1.3	75.7±2.9	53.5±2.1
DS-GNN (GRAPHCONV) (EGO)	89.4±5.4	66.6±6.5	76.7±5.4	81.3±1.9	79.6±2.0	76.6±4.0	53.1±1.5
DS-GNN (GRAPHCONV) (EGO+)	90.4±5.8	67.4±4.7	76.8±4.3	82.8±2.5	80.6±1.3	76.0±1.6	53.3±2.4
DSS-GNN (GRAPHCONV) (ED)	91.0±5.8	66.3±7.7	75.7±3.6	83.1±2.3	82.9±1.0	75.8±2.8	**53.7**±2.8
DSS-GNN (GRAPHCONV) (ND)	90.6±5.2	65.4±5.8	76.2±5.0	83.7±1.7	82.4±1.3	75.1±3.2	53.3±2.6
DSS-GNN (GRAPHCONV) (EGO)	91.5±4.9	68.0±6.1	76.6±4.6	83.5±1.1	82.5±1.6	76.3±3.6	53.1±2.8
DSS-GNN (GRAPHCONV) (EGO+)	**92.0**±5.0	67.7±5.7	77.0±5.4	83.4±1.8	82.6±1.5	76.6±2.8	53.6±2.8

Experiments

Table 2: Test results for OGB datasets. Gray background indicates that ESAN outperforms the base encoder.

Method	OGBG-MOLHIV ROC-AUC (%)	OGBG-MOLTOX21 ROC-AUC (%)
GCN (Kipf & Welling, 2017)	76.06±0.97	75.29±0.69
DS-GNN (GCN) (ED)	74.70±1.94	74.86±0.92
DS-GNN (GCN) (ND)	74.40±2.48	75.79±0.30
DS-GNN (GCN) (EGO)	74.00±2.38	75.41±0.72
DS-GNN (GCN) (EGO+)	73.84±2.58	74.74±0.96
DSS-GNN (GCN) (ED)	76.00±1.41	75.34±0.69
DSS-GNN (GCN) (ND)	75.17±1.35	75.56±0.59
DSS-GNN (GCN) (EGO)	76.16±1.02	76.14±0.53
DSS-GNN (GCN) (EGO+)	76.50±1.38	76.29±0.78
GIN (Xu et al., 2019)	75.58±1.40	74.91±0.51
DS-GNN (GIN) (ED)	76.43±2.12	75.12±0.50
DS-GNN (GIN) (ND)	76.19±0.96	75.34±1.21
DS-GNN (GIN) (EGO)	78.00±1.42	76.22±0.62
DS-GNN (GIN) (EGO+)	77.40±2.19	76.39±1.18
DSS-GNN (GIN) (ED)	77.03±1.81	76.71±0.67
DSS-GNN (GIN) (ND)	76.63±1.52	77.21±0.70
DSS-GNN (GIN) (EGO)	77.19±1.27	77.45±0.41
DSS-GNN (GIN) (EGO+)	76.78±1.66	77.95±0.40

- Introduction
- Equivariant Subgraph Aggregation Networks(ESAN)
- A WL Analogue for ESAN
- Experiments
- Summary

Summary

- The core idea is to learn the subgraphs set instead of original graph.

Summary

- The core idea is to learn the subgraphs set instead of original graph.
- The DSS-GNN framework take 3x the time of the corresponding base graph encoder to obtain a little promotion.