# Improving Subgraph Representation Learning via Multi-View Augmentation

**Anonymous Authors**[1]

## Abstract

Subgraph representation learning based on Graph Neural Network (GNN) has broad applications in chemistry and biology, such as molecule property prediction and gene collaborative function prediction. On the other hand, graph augmentation techniques have shown promising results in improving graph-based and node-based classification tasks but are rarely explored in the GNN-based subgraph representation learning literature. In this work, we developed a novel multiview augmentation mechanism to improve subgraph representation learning and thus the accuracy of downstream prediction tasks. The augmentation technique creates multiple variants of subgraphs and embeds these variants into the original graph to achieve both high training efficiency, scalability, and improved accuracy. Experiments on several real-world subgraph benchmarks demonstrate the superiority of our proposed multi-view augmentation techniques.

## 1. Introduction

Subgraph representation learning based on Graph Neural Networks (GNN) has broad applications in chemistry and biology, such as molecule property prediction and gene collaborative function prediction. For example, in protein–protein interaction (PPI) (Marinka Zitnik & Leskovec, 2018) network where a set of proteins can collaboratively determine a biological process or cellular functions, proteins and their interactions are treated as nodes and edges, respectively. A subgraph and the corresponding property is the set of collective proteins and their biological functions. Moreover, in a molecular cluster, atoms and bonds can be viewed as nodes and edges respectively. Any particular molecule in the system is represented as a subgraph of the molecular cluster.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

On the other hand, graph augmentation approaches have shown promising results in promoting the expressive power of GNNs in graph-level and node-level tasks. To name a few, (You et al., 2020) dropped nodes to build perturbed graphs for contrastive learning. (Rong et al., 2019) imported the dropping edge strategy to perturb random edges to avoid over-fitting and relieve over-smoothing when training GNNs. Although several subgraph-based GNNs have been proposed to learn subgraph representations for subgraph-related tasks (Alsentzer et al., 2020; Kim & Oh, 2022; Wang & Zhang, 2021), none of them considers graph augmentation techniques to improve their accuracy.

In this work, we raise a fundamental question: *Can graph augmentation strategies boost subgraph representation learning?* To answer the question, we applied general graph augmentation techniques such as dropedge (You et al., 2020) to existing subgraph-based GNNs and observed surprisingly significant accuracy degradation. We see that the performance on HPO-METAB dropped 21.4% when applying the dropedge strategy on subgraphs, and when applying the existing methods directly on the whole-graph and subgraph perturbation, the Micro-F1 score dropped over 5% on average. This motivated us to develop subgraph-specific augmentation techniques to improve subgraph representation learning.

To this end, we present a novel multi-view approach to augment subgraph representation learning. Our basic idea is to not only create perturbations of the original subgraphs but also preserve the original view of the subgraph during training. Instead of directly generating two separate graphs containing different views of the same subgraph, this work takes another strategy to significantly decrease the demand for GPU resources– that is, generating stochastic augmented views when feeding forward a certain batch of data. Feeding the augmented subgraph into the subgraph-specific neural networks, we can get the embeddings of both original and augmented subgraphs. Then, we can generate subgraph embeddings by applying a pooling function.

In summary, this work makes the following contributions:

- This work proposes a novel multi-view augmentation strategy to improve the accuracy of subgraph-based learning tasks. This is the first work that explores the

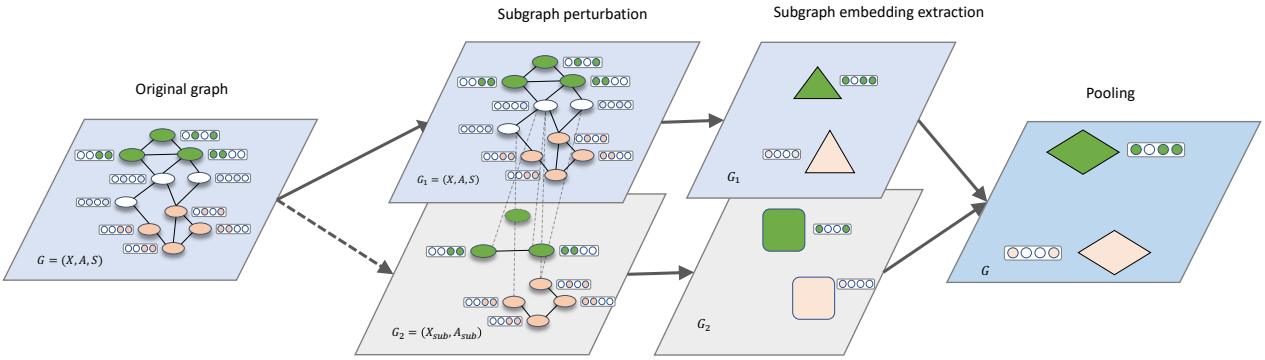


Figure 1: Overview of our proposed subgraph augmentation approach. With the original partition of subgraphs (green and orange nodes), we first generate multi subgraph views via stochastic augmentation such as dropping nodes and edges. Then, the augmented subgraph will be connected to the remaining part of the original graph, which is, adding edges that link the augmented subgraph and whole graph. After feeding forward the whole graph into subgraph-specific GNNs, we can extract the subgraph embeddings of different views, respectively (triangles and squares). Ultimately, the embeddings of different views are fused by a pooling function and thus we have the augmented subgraph embeddings (diamonds).

benefits of graph augmentation techniques in subgraph representation learning.

- The proposed multi-view augmentation strategy dynamically binds augmented subgraph views to the whole graph to decrease exaggerated GPU resource consumption in order to achieve both high training efficiency and improved task accuracy.

- Empirical evaluations on four subgraph datasets demonstrate that our augmentation approach can improve existing subgraph representation learning by 0.1%-1.6% in accuracy, 10% higher than general graph augmentation techniques dropedges and dropnodes.

## 2. Related Works

### 2.1. Subgraph Representation Learning

Subgraph neural network learning is gaining more and more attentions these years (Meng et al., 2018). SubGNN (Alsentzer et al., 2020) routes messages for internal and border properties within sub-channels for each channel: neighborhood, structure, and position. After that, the anchor patch is sampled and the features of the anchor patch are aggregated to the connected components of the subgraph by six sub-channels. GLASS (Wang & Zhang, 2021) employs labeling trick (Zhang et al., 2021) and labels nodes belonging to any subgraph to boost plain GNNs on subgraph tasks. S2N (Kim & Oh, 2022) translates subgraphs into nodes thus cuts the scale of the input graph. None of these approaches leverages graph augmentations to improve task accuracy.

### 2.2. Graph Augmentation

DGI (Veličković et al., 2018) perturbs the nodes by performing a row-wise swap of the input feature matrix while the adjacency matrix remains unchanged, which generates negative samples for comparison learning and maximizes the mutual information of input and output. GAug-M (Ding et al., 2022) generates and removes edges of the graph by training an edge predictor to finally achieve the effect of high connectivity between nodes within the same class and low connectivity between nodes from different classes. NeuralSparse (Zheng et al., 2020) proposes a supervised graph sparsification technique that improves generalization by learning to remove potentially task-irrelevant edges from the input graph. (You et al., 2020) points out that different data augmentation techniques introduce different advantages in graph learning tasks in different domains. For example, edge perturbation can be good for enhancing learning in social network graphs, but in compound graphs learning, edge perturbation can be counterproductive by destroying the original information. SUBG-CON (Jiao et al., 2020) samples a series of subgraphs containing regional neighbors from the original graph as training data to serve as an augmented node representation.

## 3. Method

### 3.1. Preliminaries

Let $G = (\mathcal{V}, \mathcal{E}, \mathbf{X})$ denote a graph, where $\mathcal{V} = \{1, 2, .., N\}$ represents the node set, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ represents the edge sets, and $\mathbf{X}$ represents the corresponding node feature. $X_i$, the $i^{th}$ row of $\mathbf{X}$ represents the features associated with the $i^{\text{th}}$ node and the adjacency matrix $\mathbf{A} \in \{0, 1\}^{N \times N}$ where $a_{ij} = 1$ denotes that $(v_i, v_j) \in \mathcal{E}$. $S = (\mathcal{V}_\mathcal{S}, \mathcal{E}_\mathcal{S}, \mathbf{X_S})$ denotes a

subgraph of $G$, where $\mathcal{V}_\mathcal{S} \subseteq \mathcal{V}$, $\mathcal{E}_\mathcal{S} \subseteq \mathcal{E} \cap (\mathcal{V}_\mathcal{S} \times \mathcal{V}_\mathcal{S})$ and $\mathbf{X_S}$ stacks the rows of $\mathbf{X}$ belong to $\mathcal{V}_\mathcal{S}$.

Given the subgraphs $\mathcal{S} = (S_1, S_2, .., S_n)$, and their labels $\mathbf{T} = \{t_{S_1}, t_{S_2}, ..., t_{S_n}\}$, the problem of *subgraph property prediction* is to learn a representation embedding $h_S$ to predict the corresponding $t_{S_i}$.

### 3.2. Proposed Multi-View Augmentation

This section presents our proposed multi-view augmentation approach (illustrated in Figure 1). At each forwarding step, we first generate new views of subgraphs in this batch and then perturb them randomly. After that, we apply concatenation on node sets and adjacency matrix to generate a new whole graph. Finally, we pass them into a subgraph-specific neural network and pool the corresponding embeddings to get the augmented subgraph embeddings.

#### 3.2.1. Drop Nodes

We generate a stochastic boolean mask $V_p$, whose size is identical to the subgraph node number, where $V$ is the total number of the subgraph nodes in the current batch and $p$ is the rate of dropping nodes. Thus, the corresponding adjacency matrix becomes

$$A_{s_{\text{drop}}} = A_s - A'_s \tag{1}$$

where $A'_s$ is the sparse matrix inferred by $V_p$ whose rows and columns where $V_p \neq 1$ will be set to zero. Thus, it randomly selects several nodes to mask and masks the corresponding rows and columns in the original subgraph adjacency matrix.

#### 3.2.2. Drop Edges

We generate a stochastic boolean mask $E_p$, where $E$ is the size of subgraph edges in the current batch and $p$ is the rate of dropping edges.

$$A_{s_{\text{drop}}} = A_s - A'_s \tag{2}$$

where $A'_s$ is the sparse matrix expanded by $E_p$. Consequently, It randomly masks certain $a_{ij}$ where $a_{ij} = 1$ in the original subgraph adjacency matrix.

#### 3.2.3. Adapting to Original Graph

After augmentation, we have a set of new subgraph nodes, they clone the corresponding subgraph nodes in original graph nodes $\mathcal{V}$. Denoting them as $\mathcal{V}_{\text{generated}}$, the new-generated node set is formed as

$$\mathcal{V}_{\text{new}} = \begin{bmatrix} \mathcal{V} & \mathcal{V}_{\text{generated}} \end{bmatrix} \tag{3}$$

If we extract $A$ to be $A_l$, where the filtering row-index is the original subgraph nodes in $\mathcal{V}_{\text{generated}}$, the consequent adjacency matrix becomes

$$A_{\text{new}} = \begin{bmatrix} A & A_l^T \\ A_l & A_{s_{\text{drop}}} \end{bmatrix} \tag{4}$$

For each subgraph in this batch, $S = (\mathcal{V}_\mathcal{S}, \mathcal{E}_\mathcal{S}, \mathbf{X_S})$ are translated correspondingly to be

$$S_{\text{new}} = (\mathcal{V}_{\mathcal{S}\,\text{generated}}, \mathcal{E}_{\mathcal{S}\,\text{generated}}, \mathbf{X}_{\mathbf{S}\,\text{generated}}). \tag{5}$$

Thus, the consequent subgraph set becomes will be

$$S_{\text{new}} = \begin{bmatrix} S & S_{\text{generated}} \end{bmatrix} \tag{6}$$

Feeding forward the augmented subgraph into subgraph-specific neural networks, we get

$$h_{S_{\text{new}}} = \text{GNN}\left(\mathcal{V}_{\text{new}}, A_{\text{new}}, S_{\text{new}}\right) \tag{7}$$

By applying pooling functions, which fuse different subgraph embeddings into one embedding, we get

$$h_{S_{\text{augmented}}} = \text{Pool}\left(h_{S_{\text{new}}}\right) \tag{8}$$

Therefore, we are able to use functions such as a multi-layer perceptron (MLP) to predict the subgraph properties by applying classification or regression functions.

$$t_S = \text{Predict}\left(h_{S_{\text{augmented}}}\right) \tag{9}$$

## 4. Experiments

In this section, we evaluate the efficacy of our proposed augmentation technique by comparing with several baselines on real-world datasets.

### 4.1. Experiment Settings

Table 1: Statistics of four real-world datasets

| Dataset | #Nodes | #Edges | #Subgraphs |
|---|---|---|---|
| PPI_BP | 17,080 | 316,951 | 1591 |
| HPO_METAB | 14,587 | 3238,174 | 2400 |
| HPO_NEURO | 14,587 | 3238,174 | 4000 |
| EM_USER | 57,333 | 4573,417 | 324 |

**Datasets** Table 1 reports the statistics of the datasets. Specifically, PPI-BP (Marinka Zitnik & Leskovec, 2018) task is to predict the collective cellular function of a given set of genes known to be associated with certain common biological processes. The base graph is the human protein–protein interaction (PPI) network and the protein subgraph

Table 2: Mean Micro-F1 with standard error of the mean on on four real-world datasets. Results are provided with running with 10 random seeds.

| METHOD | PPI_BP | HPO_METAB | HPO_NEURO | EM_USER |
|---|---|---|---|---|
| GLASS W/ DROP NODE | $0.597 \pm 0.003$ | $0.564 \pm 0.012$ | $0.657 \pm 0.007$ | $0.814 \pm 0.032$ |
| GLASS W/ DROP EDGE | $0.593 \pm 0.007$ | $0.602 \pm 0.090$ | $0.659 \pm 0.006$ | $0.814 \pm 0.032$ |
| GLASS W/ DROP EDGE SUB | $0.575 \pm 0.007$ | $0.490 \pm 0.020$ | $0.633 \pm 0.010$ | $0.808 \pm 0.026$ |
| GLASS | $0.613 \pm 0.007$ | $0.614 \pm 0.005$ | $0.685 \pm 0.005$ | $0.888 \pm 0.006$ |
| **GLASS w/ Multi-View (Ours)** | $\mathbf{0.623 \pm 0.008}$ | $\mathbf{0.615 \pm 0.010}$ | $\mathbf{0.685 \pm 0.003}$ | $\mathbf{0.904 \pm 0.006}$ |

labels them according to cellular functions from six categories (metabolism, development, stress-death, cell-org, transport, and signal-transduction).

HPO-METAB and HPO-NEURO (Splinter et al., 2018; Hartley et al., 2020) simulate rare disease diagnostic tasks with the task of predicting subcategories of metabolic or neurological disorders that are most consistent with these phenotypes. The base graph is a knowledge graph containing phenotypic and genotypic information for rare diseases, and the subgraph consists of a collection of phenotypes associated with rare monogenic diseases. HPO-METAB subgraphs are labeled according to the type of metabolic disease, and HPO-NEURO subgraphs are labeled with one or more neurological diseases.

The task of the EM-USER dataset (Ni et al., 2019) is to predict the characteristics of a user given the exercise history of the user represented by that user's subgraph. The underlying graphs are social fitness networks in which nodes represent workouts and edges exist between workouts completed by multiple users. Each subgraph is represented as a subnetwork that constitutes the user's workout history, labeled with features about the user.

**Models** The proposed augmentation technique could be integrated on top of many subgraph-based GNNs since it doesn't change the GNN model. In the evaluation, We chose GLASS (Wang & Zhang, 2021), the current state-of-the-art subgraph representation learning model as our baseline model, and did experimental validation on it with subgraph augmentations.

**Implementation Details** We employed a GLASS with hyperparameters used in (Wang & Zhang, 2021), which is corresponding to different datasets. We also modified the dropping edges and nodes probability and the respective view number to perform the hyperparameter search.

For every experiment, we first trained the model in an unsupervised manner followed by GLASS, then, the used the resulting embeddings to participate in the following subgraph property learning in a supervised fashion. To get a more generalized conclusion, we chose to perform 10 different training and validation processes with 10 distinct random seeds.

### 4.2. Results

The empirical performance is summarized in Table 2. From the table, we can see that our proposed model shows a certain amount of promotion across all four datasets. Subgraph augmentation approaches consistently perform better than GLASS mainly because it inhibits the oversmoothing and overfitting of the original graph while it loses a small partition of information because it keeps the original graph information for models' perception.

We evaluated our approach by comparing dropping nodes and edges on whole graph, dropping edges on subgraph, and plain GLASS. However, when the graph is perturbed without multi-view augmentation, the performance falls dramatically. The strong performance of constructing multi-views verifies the superiority of the proposed multi-view augmentation framework. On HPO-METAB dataset, we see that when dropping edge directly in the subgraph, it destroys the inner connection of a subgraph so that some of the information was not maintained, thus the subgraph representation was weakened, which demonstrates that it's necessary to keep at least one view of the original subgraph structure. Meanwhile, the performance of multi-view augmentation approach does not boost that much in HPO-METAB and HPO-NEURO datasets. This subgraph augmentation strategy may be too simple for large-scale graphs which contain rich and diverse subgraphs.

## 5. Conclusion

This paper presented a novel multi-view augmentation technique to improve subgraph representation learning. The augmentation randomly perturbs the subgraph components in an input graph and generates multi-views of this subgraph. Existing subgraph-based GNNs can be used to produce embeddings for each view of the subgraph. The embeddings are fused by a pooling function to create a final representation of the subgraph. We empirically evaluate the augmentation technique on top of GLASS, a state-of-the-art subgraph-based GNN, and demonstrate the augmentation can effectively improve the accuracy of the GNN.

## References


Alsentzer, E., Finlayson, S., Li, M., and Zitnik, M. Subgraph neural networks. *Advances in Neural Information Processing Systems*, 33:8017–8029, 2020.

Ding, K., Xu, Z., Tong, H., and Liu, H. Data augmentation for deep graph learning: A survey, 2022.

Hartley, T., Lemire, G., Kernohan, K. D., Howley, H. E., Adams, D. R., and Boycott, K. M. New diagnostic approaches for undiagnosed rare genetic diseases. *Annual review of genomics and human genetics*, 21:351–372, 2020.

Jiao, Y., Xiong, Y., Zhang, J., Zhang, Y., Zhang, T., and Zhu, Y. Sub-graph contrast for scalable self-supervised graph representation learning. In *2020 IEEE International Conference on Data Mining (ICDM)*, pp. 222–231. IEEE, 2020.

Kim, D. and Oh, A. Efficient representation learning of subgraphs by subgraph-to-node translation. *arXiv preprint arXiv:2204.04510*, 2022.

Marinka Zitnik, Rok Sosič, S. M. and Leskovec, J. BioSNAP Datasets: Stanford biomedical network dataset collection. http://snap.stanford.edu/biodata, August 2018.

Meng, C., Mouli, S. C., Ribeiro, B., and Neville, J. Subgraph pattern neural networks for high-order graph evolution prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Ni, J., Muhlstein, L., and McAuley, J. Modeling heart rate and activity data for personalized fitness recommendation. In *The World Wide Web Conference*, pp. 1343–1353, 2019.

Rong, Y., Huang, W., Xu, T., and Huang, J. Dropedge: Towards deep graph convolutional networks on node classification. *arXiv preprint arXiv:1907.10903*, 2019.

Splinter, K., Adams, D. R., Bacino, C. A., Bellen, H. J., Bernstein, J. A., Cheatle-Jarvela, A. M., Eng, C. M., Esteves, C., Gahl, W. A., Hamid, R., et al. Effect of genetic diagnosis on patients with previously undiagnosed disease. *New england journal of medicine*, 379(22):2131–2139, 2018.

Veličković, P., Fedus, W., Hamilton, W. L., Liò, P., Bengio, Y., and Hjelm, R. D. Deep graph infomax. *arXiv preprint arXiv:1809.10341*, 2018.

Wang, X. and Zhang, M. Glass: Gnn with labeling tricks for subgraph representation learning. In *International Conference on Learning Representations*, 2021.

You, Y., Chen, T., Sui, Y., Chen, T., Wang, Z., and Shen, Y. Graph contrastive learning with augmentations. *Advances in Neural Information Processing Systems*, 33: 5812–5823, 2020.

Zhang, M., Li, P., Xia, Y., Wang, K., and Jin, L. Labeling trick: A theory of using graph neural networks for multi-node representation learning. *Advances in Neural Information Processing Systems*, 34, 2021.

Zheng, C., Zong, B., Cheng, W., Song, D., Ni, J., Yu, W., Chen, H., and Wang, W. Robust graph representation learning via neural sparsification. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 11458–11468. PMLR, 2020.