MoLGNN: Self-Supervised Motif Learning Graph Neural Network for Drug Discovery

Xiaoke Shen*
Ph.D. Program in Computer Science, The Graduate Center, The City University of New York
New York, NY 10016, USA
xshen@gradcenter.cuny.edu

Yang Liu*
Department of Computer Science, Hunter College, The City University of New York
New York, NY 10065, USA
yl1708@hunter.cuny.edu

You Wu
Ph.D. Program in Computer Science, The Graduate Center, The City University of New York
New York, NY 10016, USA
ywu1@gradcenter.cuny.edu

Lei Xie
Ph.D. Program in Computer Science, The Graduate Center, The City University of New York
Department of Computer Science, Hunter College, The City University of New York
Helen and Robert Appel Alzheimer's Disease Research Institute, Weill Cornell Medicine
New York, NY 10065, USA
lxie@iscb.org

*** Equal contributions**

Abstract

AI-facilitated drug design has become a popular field that has the potential to accelerate drug discovery dramatically. Generally speaking, training a generalized deep learning model needs a large amount of correctly labeled data. However, since it is usually much harder to label chemicals than images or natural languages, lack of high-quality training samples is often a critical barrier in AI-based drug screening. In this work, we have developed a self-supervised Motif Learning Graph Neural Network (MoLGNN) that can significantly improve the performance of computational drug screening by exploiting unlabeled chemicals to pretrain the model. MoLGNN conducts self-learning by 1), utilizing the node features and network motifs of a graph as self-generated labels and 2), reconstructing edges using a Graph Isomorphism Network Variational Auto-Encoder (GINVAE). These two strategies are combined in a multi-task learning framework. Our experiment results show MoLGNN achieves state-of-the-art performance in multiple benchmark datasets, and is robust when the number of labeled data is small. Furthermore, we have applied MoLGNN to screen drugs for JAK1/2/3, which are novel drug targets for treating cytokine storm, a life-threatening syndrome of COVID-19. MoLGNN can be leveraged to various machine learning tasks in chemistry when high quality labeled data is scarce.

Machine Learning for Molecules Workshop at NeurIPS 2020. https://ml4molecules.github.io

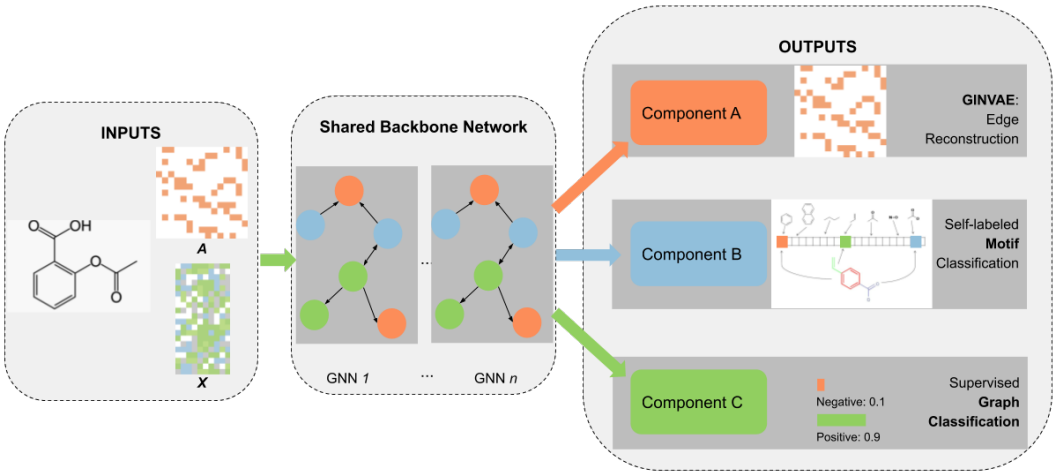

Figure 1: **Overview of the whole system**: The input is chemicals represented by the adjacency matrix ($\mathbf{A}$) and node attributes ($\mathbf{X}$). We have a shared GNN backbone network followed by three components for edge reconstruction, Motif learning, and graph classification. For more details, please check the model architecture session.

1 Introduction

Graph Neural Networks (GNNs) have received tremendous attention over the past few years because of their success in handling non-Euclidean structured graph data [22, 11, 3, 19, 4]. The early application of GNNs mainly focused on relation prediction, e.g. co-purchased products. Recently, GNNs have been extended to the naturally graph-structured chemical compounds to expedite property prediction and drug discovery [6, 12, 2, 13]. If a computational model can fully or partially replace the costly drug screening, it will undoubtedly facilitate drug discovery process.

The difficulty in applying GNN model to drug discovery is the insufficiency of labeled data. Nevertheless, generating a large amount of labeled chemical data is challenging. It needs a certain level of expertise and large-scale experimental screening. The sparsity of labeled data will cause inaccurate prediction for new chemicals that are not similar to the training data [1, 14]. Thus, using unlabeled data to pretrain a GNN model becomes a popular method to improve the robustness of GNN models by making the latent space of GNN model less sparse [6, 8, 7]. Furthermore, pretraining also improves model performance in imbalanced datasets [5]. Most of previous GNN pretraining methods focus on either node-level or graph-level information [8, 7]. However, these methods fail to learn information of substructures in a graph, especially edge attributes and node-edge connection information, which is important for precise prediction of properties of chemical compounds.

In this paper, we introduce a new graph pretraining approach with a focus on chemical structures. The core of our method is a self-supervised motif learning GNN (MoLGNN). Given a graph, its network motifs, which are recurrent substructures of a graph, can be self-labeled by whether or not they exist in the graph. The MoLGNN is further pretrained using the motif as a label in the framework of multi-task supervised learning. Extensive benchmark studies demonstrate that MoLGNN achieves state-of-the-art performance in predicting chemical properties, such as substrate binding, toxicity, and inhibitory effects. Moreover, MoLGNN is robust even with significantly few labeled data.

To demonstrate the utility of MoLGNN, we test MoLGNN in screening drugs that are targeting Janus kinase (JAK), which is a family of intracellular tyrosine kinases that are critical components in transmitting cytokine signals. JAKs are considered a novel target for the treatment of COVID-19 [15], which has caused a global pandemic. The primary lethal syndrome of COVID-19 is cytokine storm, an acute immune response that results in overdosed cytokines release into the blood in a short range of time. Inhibiting the activity of JAKs, thus, blocking the cytokine signaling pathway, is an efficient way to alleviate body responses to cytokine storms. With only 10% data as the training set, our model can achieve over 89% AP-AUC and ROC-AUC when predicting the binding profile of

Table 1: **Test ROC-AUC (%) performance on molecular prediction benchmarks.** *The same simple atom features as ContextPred are used for a fair comparison.

Dataset	BACE	BBBP	SIDER	ClinTox	HIV	Average
ContextPred w/o pretrain	70.1 ±5.4	65.8 ±4.5	57.3 ±1.6	58.0 ±4.4	75.3 ±1.9	65.3
ContextPred	84.5 ±0.7	68.7 ±1.3	62.7 ±0.8	72.6 ±1.5	**79.9 ±0.7**	73.9
Ratio of improvement	**1.20**	1.04	1.09	1.25	1.06	1.13
Non MoLGNN*	71.1 ±1.3	66.3 ±1.0	59.1 ±0.5	55.2 ±0.8	66.0±1.8	63.5
GINVAE only*	80.8 ±0.6	70.2 ±0.8	62.3 ±0.2	73.4 ±1.1	70.0 ±0.9	71.3
Motif only*	79.2 ±1.1	68.9 ±1.0	64.0 ±0.3	72.4 ±1.3	74.9±0.8	71.9
MoLGNN*	84.0 ±0.5	69.4 ±1.5	**64.1 ±0.5**	73.8 ±1.1	76.8±0.1	73.6
Ratio of improvement	1.18	**1.05**	1.09	**1.34**	**1.16**	**1.16**
Non MoLGNN (ours)	77.8 ±6.0	84.8 ±0.4	56.9 ±0.3	88.3 ±0.3	74.1 ±1.1	76.4
GINVAE only(ours)	87.1 ±1.5	**89.2 ±1.6**	61.7 ±1.0	93.7 ±0.0	76.3 ±0.2	81.6
Motif only(ours)	85.3 ±1.6	86.1 ±0.5	61.5 ±1.7	93.7 ±0.1	77.2 ±0.7	80.7
MoLGNN (ours)	**87.4 ±1.7**	88.9 ±1.1	63.6 ±0.3	**94.2 ±0.2**	78.0 ±0.0	**82.4**
Ratio of improvement	1.12	**1.05**	**1.18**	1.07	1.05	1.09

drug-like chemicals. It can potentially accelerate the drug discovery and development process for COVID-19.

2 Methods

Problem Formulation Let $G = (V, E)$ be a graph with $N = |V|$ nodes and $M = |E|$ edges. Given a molecule with N atoms and their atomic numbers $\boldsymbol{Z} = \{Z_1, \ldots, Z_N\}$ as well as M bonds, a graph G is constructed such that the atom is node and the bond is edge. The problem of molecular property prediction is to predict the target property $t \in \mathbb{C}$ of the molecule. The classification goal is to find a function $f : \{\boldsymbol{Z}\} \to \mathbb{C}$. When given auxiliary chemical information such as atomic features $\boldsymbol{\Theta}$, the goal function is $f : \{\boldsymbol{Z}, \boldsymbol{\Theta}\} \to \mathbb{C}$.

Network Motif Network motif is the recurrent substructure or sub-graph in a graph. In a chemical compound, chemical functional groups or fragments such as benzene rings are naturally occurred motifs. In this work, we use PubChem fingerprints to represent the motif. The PubChem fingerprints used in pretraining are calculated with the Chemistry Development Kit [20] and a Python interface package PyFingerprint [9]. The original fingerprint has 881 digits representing atoms, bonds, and substructures in a chemical compound. In our work, because the chemicals in the datasets are majorly organic drugs, we reduce the number of digits of the fingerprint by removing the digits associate to atoms that are less likely to appear in drugs. Specifically, only the digits related to C, H, O, N, S, F, Cl, and Br atoms are kept, which results in a filtered fingerprint with 740 digits. It is worth noting that, in chemical compounds, substructures related to chemical reactions are termed moieties. In our work, the GNN model learns substructures beyond the scope of chemical moieties. To be consistent with the terminology of general graphs, we will use the motif throughout the paper to refer to the substructure of chemical compounds.

Model Architecture As shown in Figure 1, the model is constructed in a multi-task learning framework with three tasks: edge reconstruction of molecular graph, self-supervised motif learning, and supervised graph classification. Details see appendix.

Pretrain Procedure We train four classification networks for each task: 1) *Non MoLGNN*: no pretraining is used, and the network is trained with the standard supervised classification approach. 2) *GINVAE only*: the network is pretrained by GINVAE, and then fine-tuned by gradually unfreezing the weights of the shared network. 3) *Motif only*: the network is pretrained by Motif learning network, and then fine-tuned by gradually unfrozen the weights of shared network. 4) *MoLGNN*: the network is pretrained by both GINVAE and Motif learning network and then fine-tuned.

Table 2: **JAK 1/2/3 ROC-AUC(%) and Average Precision(%) performance under Scaffold Splitting**

Dataset	JAK 1		JAK 2		JAK 3	
	ROC	AP	ROC	AP	ROC	AP
Non MoLGNN	91.8±0.6	85.4±0.7	87.2±0.3	85.5±0.8	81.2±2.5	80.2±2.0
GINVAE Only	94.2±0.1	88.6±0.2	88.0±0.9	87.3±0.4	84.6±0.2	84.7±0.8
Motif Only	93.4±0.4	87.8±1.1	85.8±0.9	87.5±0.2	86.1±0.1	86.7±0.4
MoLGNN	**94.5±0.3**	**89.2±0.5**	**89.6±0.2**	**89.9±0.4**	**89.2±0.2**	**89.5±0.2**

Datasets and Data Splitting In our experiments, we use two groups of datasets. To compare with previous state-of-the-art techniques, we conduct the same experiments in BACE, BBBP, ClinTox, HIV, and SIDER datasets from MoleculeNet [21]. To test the efficiency of our model in facilitating drug development, we use datasets including kinases JAK1, JAK2, and JAK3, which are originated from chemical binding screening and are manually annotated in ChEMBL database [16]. Chemicals with IC50 value less than 10 μM are labeled as positive while others are labeled as negative. The JAK1, JAK2, and JAK3 contain 3717, 5853, and 3520 drug-like chemicals, respectively. In our experiments, we split the datasets with scaffold splitting. See appendix for details.

3 Results

The baseline method which we choose to compare with is the current state-of-the-art implementation, ContextPred [6], in which the GNN model is pretrained on the node-level with a large amount of unlabeled data and graph-level with labeled data [6]. Table 1 shows that we achieve a new state-of-the-art performance in BACE, SIDER, and ClinTox Datasets. For the HIV dataset, although our final result is moderately behind Hu et al.'s work, it is worth noting that the improvement gained from pretraining in Hu et al.'s work is 4.6 while ours is 10.8. For the BACE dataset, our result is competitive.

One challenge of working with chemical data is the scarcity of labeled data. Thus, we test if our method can still reach a good performance when using less labeled fine-tuning data. We find that, with less labeled fine-tuning data, the improvement of the pretrained model over the model without pretraining is even larger. Specifically, in the BACE dataset, MoLGNN shows a relative improvement of 12.7% over the Non-MoLGNN when 10% data are labeled, while this improvement reduces to 7.2% when using 90% labeled data.

Our technique has the potential to enhance the real-world drug discovery process. To prove that, we introduce the JAK dataset, which contains the binding affinity to three kinases JAK1, JAK2, and JAK3 for over 10,000 drug-like chemicals. We test if our model can correctly predict whether a chemical serves as a substrate of JAKs. The experiment results are shown in Table 2. From the results, we can see a significant performance boosting with the MoLGNN when using scaffold splitting benchmark where the chemicals in the testing set are structurally different from those in the training set.

4 Conclusion

We develop a new self-supervised learning method to learn improved chemical embeddings with the GNN. Comparing to the state-of-the-art technique, our method has following advantages: 1) In the pretraining step, MoLGNN is fully self-supervised. It does not need extra labeled data to get graph-level embeddings. 2) Our method not only captures atom-atom connection information, but also acquires the substructure information. Effectively capturing these information is crucial for the robust performance in chemical related tasks.3) Our method is good at handling sparse data by utilizing the target chemicals as pretraining inputs. The GNN model trained with our method shows a high level of robustness to small amount of labeled fine-tuning data. Even with very few fine-tuning data, the pretraining can still improve the classification performance by a large margin. This observation validates that our method captures features of chemical compounds very well in its latent space. The extension of our work could be replacing the GIN with more advanced GNN models. It would be interesting to find out if adding other components to our method will further strengthen the pretrained model and study why fingerprint works better than the masking method [6].

References

[1] Gediminas Adomavicius and Jingjing Zhang. Stability of collaborative filtering recommendation algorithms, 2010.

[2] Gary Bécigneul, Octavian-Eugen Ganea, Benson Chen, Regina Barzilay, and Tommi Jaakkola. Optimal transport graph neural networks, 2020.

[3] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems (NeurIPS) 29*, pages 3844–3852. Curran Associates, Inc., 2016.

[4] William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs, 2017.

[5] Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty, 2019.

[6] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. Strategies for Pre-training Graph Neural Networks. *arXiv e-prints*, art. arXiv:1905.12265, May 2019.

[7] Ziniu Hu, Changjun Fan, Ting Chen, Kai-Wei Chang, and Yizhou Sun. Pre-training graph neural networks for generic structural feature extraction, 2019.

[8] Ziniu Hu, Yuxiao Dong, Kuansan Wang, Kai-Wei Chang, and Yizhou Sun. Gpt-gnn: Generative pre-training of graph neural networks, 2020.

[9] Hongchao Ji, Hanzi Deng, Hongmei Lu, and Zhimin Zhang. Predicting a molecular fingerprint from an electron ionization mass spectrum with deep neural networks. *Analytical Chemistry*, 92(13):8649–8653, Jul 2020. ISSN 0003-2700. doi: 10.1021/acs.analchem.0c01450. URL `https://doi.org/10.1021/acs.analchem.0c01450`.

[10] Thomas N. Kipf and Max Welling. Variational Graph Auto-Encoders. *arXiv e-prints*, art. arXiv:1611.07308, November 2016.

[11] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks, 2016.

[12] Johannes Klicpera, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs, 2020.

[13] Junying Li, Deng Cai, and Xiaofei He. Learning graph-level representation for drug discovery, 2017.

[14] Xiang Li, Charles X. Ling, and Huaimin Wang. The convergence behavior of naive bayes on large sparse datasets. *ACM Trans. Knowl. Discov. Data*, 11(1), July 2016. ISSN 1556-4681. doi: 10.1145/2948068. URL `https://doi.org/10.1145/2948068`.

[15] Puja Mehta, Coziana Ciurtin, Marie Scully, Marcel Levi, and Rachel C. Chambers. Jak inhibitors in covid-19: need for vigilance regarding increased inherent thrombotic risk. *European Respiratory Journal*, 2020. ISSN 0903-1936. doi: 10.1183/13993003.01919-2020. URL `https://erj.ersjournals.com/content/early/2020/07/02/13993003.01919-2020`.

[16] David Mendez, Anna Gaulton, A Patrícia Bento, Jon Chambers, Marleen De Veij, Eloy Félix, María Paula Magariños, Juan F Mosquera, Prudence Mutowo, Michał Nowotka, María Gordillo-Marañón, Fiona Hunter, Laura Junco, Grace Mugumbate, Milagros Rodriguez-Lopez, Francis Atkinson, Nicolas Bosc, Chris J Radoux, Aldo Segura-Cabrera, Anne Hersey, and Andrew R Leach. ChEMBL: towards direct deposition of bioassay data. *Nucleic Acids Research*, 47 (D1):D930–D940, 11 2018. ISSN 0305-1048. doi: 10.1093/nar/gky1075. URL `https://doi.org/10.1093/nar/gky1075`.

[17] B. Ramsundar, P. Eastman, P. Walters, and V. Pande. *Deep Learning for the Life Sciences: Applying Deep Learning to Genomics, Microscopy, Drug Discovery, and More*. O'Reilly Media, 2019. ISBN 9781492039839. URL `https://books.google.com/books?id=tYFKuwEACAAJ`.

[18] Xiaoke Shen and Ioannis Stamos. Frustum voxnet for 3d object detection from rgb-d or depth images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[19] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks, 2017.

[20] Egon L. Willighagen, John W. Mayfield, Jonathan Alvarsson, Arvid Berg, Lars Carlsson, Nina Jeliazkova, Stefan Kuhn, Tomáš Pluskal, Miquel Rojas-Chertó, Ola Spjuth, Gilleain Torrance, Chris T. Evelo, Rajarshi Guha, and Christoph Steinbeck. The chemistry development kit (cdk) v2.0: atom typing, depiction, molecular formulas, and substructure searching. *Journal of Cheminformatics*, 9(1):33, Jun 2017. ISSN 1758-2946. doi: 10.1186/s13321-017-0220-4. URL `https://doi.org/10.1186/s13321-017-0220-4`.

[21] Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay Pande. Moleculenet: A benchmark for molecular machine learning, 2017.

[22] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. A Comprehensive Survey on Graph Neural Networks. *arXiv e-prints*, art. arXiv:1901.00596, January 2019.

[23] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How Powerful are Graph Neural Networks? *arXiv e-prints*, art. arXiv:1810.00826, October 2018.

5 Appendix

Model Architecture The input of our model is chemicals represented by adjacency matrix ($\mathbf{A}$), which represents chemical bonds, and node attributes ($\mathbf{X}$). To make the comparison fair, we use only atom type and atom chirality in node attributes (simple atom feature) when comparing to ContextPred [6]. To get better performance, in JAK binding prediction task, we add atom degree, atom formal charge, hybridization type, and aromatic label to atom attributes.

The backbone network is a GNN. We use 5-layer GIN [23] with 32 hidden units in our experiments since GIN outperforms other GNN architectures [6]. Each GIN layer of the backbone network can generate learned node embeddings. Those node embeddings can be fed to different components shown in Figure 1 to achieve a multi-task learning, they are A: edge reconstructions, B: self-labled Motif classification and C: Supervised Graph Classification.

The edge reconstruction network is a Variational Graph Auto-Encoder [10] based on the GIN backbone network above. We name this sub-network Variational Graph Isomorphism Network Auto-Encoder (**GINVAE**). Edge reconstruction is a node-level task, so we can directly use the learned node embeddings. However, for the graph-level task networks used for **Motif** learning and **Graph Classification**, we have to use a READOUT function to aggregate node embeddings to graph embeddings before moving forward. We use a "MEAN" READOUT function in our experiments. We further use a two-layer MLP for each sub-network to change the embedding (node-level or graph-level) dimensions into the desired ones depending on the tasks. These MLPs are applied to each GIN layer. The **GINVAE**'s two-layer MLP has dimensions 32 and 64. The **Motif** learning network's two-layer MLP has dimensions 370 and 740 (the dimension of the PubChem fingerprints). And for the **Graph Classification** network, the hidden units in the first layer are half of the embedding vector's dimension, and the dimension of the output layer varies from different tasks. Since the MLP is applied to each GIN layer, we use a "SUM" function as a layer-level aggregation to have the desired intermediate output. Based on this output, we use the same decoder as [10] to reconstruct the graph edges. For **Motif** learning, we apply a sigmoid function to the 740-dimension outputs. A softmax (for multiple class classification) or a sigmoid layer (for binary classification or multiple label classification) will be applied to this output to generate the final supervised **Graph Classification** results.

Loss function For the GINVAE, we use the same loss function as [10]. We treat the PubChem fingerprints in the Motif learning network as a multi-label prediction problem, and a binary cross-entropy loss is used for this network. For the Graph Classification, we use the cross-entropy loss for multi-class classification and binary cross-entropy loss for binary classification or multi-label classification. For the MoLGNN pretraining, which involves two sub-network training processes, we use a combined loss [18]:

$$L_{MoLGNN}^{pretrain} = \lambda_1 L_{GINVAE} + \lambda_2 L_{Motif}$$

where L_{GINVAE} and L_{Motif} are the pretrain loss for the GINVAE and Motif learning network, respectively. We set both the λ_1 and λ_2 as 0.5.

Data splitting In our experiments, we split the datasets with scaffold splitting as in [17]. The datasets are split into training, validating, and testing sets with the ratio of 8:1:1 based on chemical scaffolds . Briefly, the Murcko scaffold of each chemicals are captured with RDKit, and only the chemicals with the same scaffold are grouped together. Then the whole group of chemicals with the same scaffold are added into only one of the training, validating, or testing set. Therefore, the testing set only contains chemicals with different scaffold from training and validating sets. The scaffold splitting forces the distribution of chemical properties different in training and testing sets and makes it difficult to get a good prediction performance with a model trained merely with labeled training data. Therefore, the splitting method could better evaluate how much the model benefits from the self-supervised pretraining with unlabeled data. More importantly, new drugs normally have different scaffolds with the existing drugs in the real production situation. With scaffold splitting, we can better understand the potential of our trained model for the drug discovery application .

In practice, one challenge of scaffold splitting is the chemicals are usually grouped into several very large groups and many small groups. The large groups are massive enough for chemicals of one group to even exceed one tenth of the whole dataset. To make sure the ratio of the training, validating, and testing set is correct, we have to add the large groups into training set. To do that, the groups are sorted by size, and as a side effect, the groups with the same amount of chemicals are sorted by the label of their first member. This drawback causes label distributions are a slightly different amoung training, validating, and testing sets, which is the case in Hu et al.'s work with default settings. To avoid uneven distribution, after sorting the groups, we randomly permute the groups with less than 5 members.