MDGNN: METAPATH-BASED DECOUPLED GRAPH NEURAL NETWORK FOR MAG240M-LSC

TECHNICAL REPORT

Huanjing Zhao*
Anhui University
hwangyeong@163.com

Yukuo Cen
Tsinghua University
cyk20@mails.tsinghua.edu.cn

Yufei He*
Beijing Institute of Technology
yufei.he@bit.edu.cn

Zhenyu Hou
Tsinghua University
houzy21@mails.tsinghua.edu.cn

Xiao Liu
Tsinghua University
shawliu9@gmail.com

Xu Cheng
Tsinghua University
chengx19@mails.tsinghua.edu.cn

ABSTRACT

Large-scale graph data have received more attention from researchers in recent years. After the successful host of the first OGB-LSC (KDD Cup 2021), the MAG240M track is retained for the second OGB-LSC at NeurIPS 2022. MAG240M is a heterogeneous graph dataset with more than 240 million nodes, including papers, authors, and institutions. In the KDD Cup 2021, the top two methods, R_UniMP and MPNN&BGRL, are message-passing-based graph neural networks, which require much time (e.g., a few GPU days) to train one model on the MAG240M dataset. However, we find that decoupled graph neural networks can obtain comparable performance with an order of magnitude less training time. Inspired by MPLP, the 4th solution of KDD Cup 2021, we propose a metapath-based decoupled graph neural network (MDGNN), which incorporates more effective features to get better results. Our solution finally gets an award-level (top-3) performance on the MAG240M track of OGB-LSC 2022.

Keywords OGB-LSC · MAG240M · Metapath · Decoupled GNNs

1 Introduction

Due to the powerful expressive power of graph structures, research on analyzing graphs with machine learning methods is gaining much attention. Researchers explore the potential relationships between nodes in large-scale graph data to satisfy realistic requirements. The OGB-LSC team (Hu et al. [2021]) organized the first large-scale graph data competition at KDD Cup 2021 with huge success. The MAG240M-LSC is reserved for the second OGB-LSC at NeurIPS 2022 competition. This dataset is extracted from the Microsoft Academic Graph (MAG) Wang et al. [2020]. The graph contains 121 million papers, 122 million authors, and 25 thousand institutions. The title and abstract of papers are fed to a RoBERTa (Liu et al. [2019]) encoder to form the 768-dimensional representations reflecting papers' semantic relations. The task of MAG240M-LSC is to predict the primary subject area of each arXiv paper. Unlike the first KDD Cup, the test set of this tack is divided into test-dev and test-challenge for the public leaderboard and the competition, while the rest splits of the data remain the same.

We analyze the winning solutions of the KDD Cup 2021 in detail. R_UniMP (Shi et al. [2021]) utilizes enhanced RGAT aggregation node features and introduces several techniques to improve the performance. The MPNN&BGRL method (Addanki et al. [2021]) adds self-supervised loss to the message passing neural network to enhance the generalization of the model. MPLP (Peng et al. [2021]) utilizes the metapath-based label propagation method, which uses less training time and memory. Considering the actual situations of our computing resources, we hope to find a light solution for the massive heterogeneous graphs. We propose a metapath-based decoupled graph neural network (MDGNN) for the

*This work was done when the author was an intern of Zhipu AI.

massive heterogeneous graph. Our solution obtains a top-3 performance (75.06% accuracy on test-challenge) on the MAG240M track of OGB-LSC 2022.

2 Methodology

In this section, we introduce our method, MDGNN, which decouples feature propagation and feature transformation. The decoupled architecture is flexible to combine features with different sources. For heterogeneous graphs, we can utilize metapath-based features to leverage heterogeneous information into our method. In the following, we introduce the details of selected features about how they are processed into vectors as model inputs. For the feature fusion step, the features are concatenated and fed into the classifier for the prediction after each feature is first transformed individually.

2.1 Feature Selection

In this part, we present and number the features used by our solution.

2.1.1 MPLP Feature

We preserve all the MPLP features in our solution since we find they are effective. **Feature #1** is from the last hidden representations of a 2-layer RGAT that stands out among all the official baselines. **Feature #2** is the original RoBERTa features of papers. **Features #3 to #14** are processed differently with label propagation. Specifically, the authors select five meta-paths and design different strategies to propagate label information.

2.1.2 Metapath2vec Feature

Metapath2vec (Dong et al. [2017]) is an effective method for learning on heterogeneous graphs, which utilizes meta-path based random walk to sample node sequences and learns representation with a heterogeneous skip-gram model. It has been shown in R_UniMP that metapath2vec brings some improvement to the model. Therefore, we consider the metapath2vec embedding as **feature #15**. The embedding is from R_UniMP solution, which is trained with 5 epochs, 3 window sizes, and 64 dimensions.

2.1.3 Three-layer RGAT Feature

In the official baseline released by OGB, the two-layer RGAT method performs well with 70.02% validation accuracy. We suggest that appropriately increasing the depth of propagation layers on a large-scale graph will learn better representation. Therefore we train a three-layer RGAT model by the number of samples 15, 25, and 10 in different layers and use the obtained node representations as **feature #16**.

2.1.4 Metapath-based Aggregation Feature

Inspired by methods that work well in the ogbn-mag dataset (e.g., Yang et al. [2022]), propagation on asymmetric meta-paths is a potential direction to explore. In this solution, we selects all meta-paths within three degrees, following SeHGNN (Yang et al. [2022]). The details of meta-paths are listed in Table 1. During the propagation process, we focus on the source and destination of the meta-path, aggregating the attributes of the destination to the source paper type nodes. The node attributes aggregated under different meta-paths constitute **features #17 to #27**.

Table 1: Meta-paths for aggregation.

Number	Meta-paths
17	Paper - cited_by - Paper - cited_by - Paper
18	Paper - cited_by - Paper - cite - Paper
19	Paper - cited_by - Paper
20	Paper - cite - Paper - cited_by - Paper
21	Paper - cite - Paper - cites - Paper
22	Paper - cite - Paper
23	Paper - writed_by - Author - cited_by - Paper
24	Paper - cited_by - Paper - writed_by - Author
25	Paper - cite - Paper - writed_by - Author
26	Paper - writed_by - Author - affiliated_with - Institution
27	Paper - writed_by - Author

2.1.5 MPNN&BGRL feature

The MPNN&BGRL solution (Addanki et al. [2021]) ranked second in the KDD Cup 2021. Different from other methods, they incorporate the auxiliary loss of the self-supervised learning algorithm BGRL (Thakoor et al. [2021]). The self-supervised learning strategy allows non-arxiv papers to be more actively integrated into the training. We select the last hidden representations of the model as **feature #28**.

2.2 Feature Fusion

We adopt a similar method with MPLP to make predictions based on different features. Given the selected features $\{H_1, H_2, ..., H_K\}$, where K means the total number of features. To project the different features into a vector space of the specified dimension, our model first maps each feature as following:

$$Z_k = f_k(H_k), \tag{1}$$

where f_k is the project function and we set them as two-layer MLPs in our solution. After obtaining the projected vector Z_k for k^{th} selected feature, we concatenate the features and feed them to the classifier for supervised learning. The final prediction is computed as follows:

$$Z = Z_1 \oplus Z_2 \oplus ... \oplus Z_K, \tag{2}$$

$$Y_{pred} = Classifier(Z), \tag{3}$$

where $\oplus$ means the concatenation operation and $Classifier(\cdot)$ is a two-layer MLP.

3 Experiments

In the previous section, we present all the candidate features used in our solution. For this section, we select various combinations of features and different parameters to train multiple models. Finally, we integrate the results of multiple models to obtain the classification results on the test-challenge set as the submission.

we conduct lots of experiments of different feature combinations. The details of the 15 models constructed with different features are shown in Table 2. All models (except model #15) are trained with five folds to obtain validation accuracy. We use an NVIDIA A100 GPU (80GB memory) and an AMD EPYC 7642 48-Core Processor (1T memory) for our experiments. For each group of 5-fold models, we need about one hour for the training, which is more efficient than message-passing-based GNNs.

Table 2: Details of all ensemble models.

Model Num	Features [Num]	Valid Acc(%)
1	MPLP [1-14]	76.69
$2 \sim 7$	MPLP [1-14] + metapath2vec [15] + meta-path based aggregation [17-27] (seeds 0,1,2,3,4,5)	76.91
8	MPLP [1-14] + metapath2vec [15] + meta-path based aggregation [17-27] (MLP hidden 1024)	76.82
9	MPLP [1-14] + metapath2vec [15] + meta-path based aggregation [17-27] (dim 128) + three-layer RGAT [16]	76.84
$10 \sim 12$	MPLP [1-14] + metapath2vec [15] + meta-path based aggregation [17-27] + three-layer RGAT [16] (seeds 0,1,2)	76.88
13	MPLP [3-14] (only label propagation) + metapath2vec [15] + three-layer RGAT [16]	76.73
14	MPLP [1-14] + metapath2vec [15] + meta-path based aggregation [17-27] + three-layer RGAT [16] + MPNN&BGRL [28]	77.01
15	MPNN&BGRL [28]	76.80

Ensemble. We finally select 15 models for the ensemble, as shown in Table 2. Models #2$\sim$#7 and Models #10$\sim$#12 come from different seeds. The model #15 is from the MPNN&BGRL model without extra training. During the process of the ensemble, we assign different weights for 15 models. To find the optimal solution for the weights, we adopt the particle swarm optimization algorithm (Poli et al. [2007]) to find a good solution. Our final accuracy results are 77.14% on the validation set, and 75.06% on the test-challenge set.

Notes. During the competition, we made a mistake in generating the representation of MPNN&BGRL on the validation set (feature #28), caused by feeding different splits of validation set for the k-fold trained MPNN&BGRL models. The mistake only influences the representations of the validation set and does not influence the test set. The artificially high validation accuracy (>79%) caused by the mistake makes the prediction of model #15 with a large weight during the ensemble process, which leads to a weak submission result of our method. Finally, we find that the correct validation accuracy of our submission is only 77.14% after we fix the bug. However, if we simply average the predictions of all 15 models, the correct accuracy on the validation set reaches 77.56%, which is better than 77.14%.

4 Conclusion

Our solution, MDGNN, adopts decoupled GNN architecture, which has less training time and memory than message-passing-based graph neural networks. We make our efforts to select effective features for node classification in the setting of massive heterogeneous graphs. In addition, we utilize different feature combinations to train multiple models for the ensemble. Finally, we sincerely thank the OGB-LSC team for the organization of the second OGB-LSC competition at NeurIPS 2022. The long-term support of OGB for the community of graph machine learning gives us opportunities to explore and gain deep understanding of the field.

References

Weihua Hu, Matthias Fey, Hongyu Ren, Maho Nakata, Yuxiao Dong, and Jure Leskovec. Ogb-lsc: A large-scale challenge for machine learning on graphs. *arXiv preprint arXiv:2103.09430*, 2021.

Kuansan Wang, Zhihong Shen, Chiyuan Huang, Chieh-Han Wu, Yuxiao Dong, and Anshul Kanakia. Microsoft academic graph: When experts are not enough. *Quantitative Science Studies*, 1(1):396–413, 2020.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Yunsheng Shi, PGL Team, Zhengjie Huang, Weibin Li, Weiyue Su, Shikun Feng, et al. R-unimp: Solution for kdd cup 2021 mag240m-lsc. *Open Graph Benchmark-Large-Scale Challenge@ KDD Cup 2021*, 2021.

Ravichandra Addanki, Peter W Battaglia, David Budden, Andreea Deac, Jonathan Godwin, Thomas Keck, Wai Lok Sibon Li, Alvaro Sanchez-Gonzalez, Jacklynn Stott, Shantanu Thakoor, et al. Large-scale graph representation learning with very deep gnns and self-supervision. *arXiv preprint arXiv:2107.09422*, 2021.

Qiuying Peng, Wencai Cao, and Zheng Pan. Metapath-based label propagation for large-scale heterogeneous graph. 2021.

Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 135–144, 2017.

Xiaocheng Yang, Mingyu Yan, Shirui Pan, Xiaochun Ye, and Dongrui Fan. Simple and efficient heterogeneous graph neural network. *arXiv preprint arXiv:2207.02547*, 2022.

Shantanu Thakoor, Corentin Tallec, Mohammad Gheshlaghi Azar, Rémi Munos, Petar Veličković, and Michal Valko. Bootstrapped representation learning on graphs. In *ICLR 2021 Workshop on Geometrical and Topological Representation Learning*, 2021.

Riccardo Poli, James Kennedy, and Tim Blackwell. Particle swarm optimization. *Swarm intelligence*, 1(1):33–57, 2007.