



HAL
open science

Molecular Activity Prediction Based on Graph Attention Network

Xiaowei Cao, Tiwen Wang, Ruohui Cheng, Jingyi Ding

► **To cite this version:**

Xiaowei Cao, Tiwen Wang, Ruohui Cheng, Jingyi Ding. Molecular Activity Prediction Based on Graph Attention Network. 5th International Conference on Intelligence Science (ICIS), Oct 2022, Xi'an, China. pp.395-401, 10.1007/978-3-031-14903-0_42 . hal-04666418

HAL Id: hal-04666418

<https://hal.science/hal-04666418v1>

Submitted on 1 Aug 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

Molecular Activity Prediction based on Graph Attention Network

Xiaowei Cao, Tiwen Wang, Ruohui Cheng, and Jingyi Ding *

International Research Center for Intelligent Perception and Computation, Joint International Research Laboratory of Intelligent Perception and Computation, School of Artificial Intelligence, Xidian University, Xi'an 710071, China
jyding@xidian.edu.cn

Abstract. Spatial convolutional models of Graph Neural Networks (GNNs) updates embeddings of nodes by the neighborhood aggregation, it has obvious advantages in reducing time complexity and improving accuracy. Therefore, it is a very important task to change the way of neighborhood aggregation to learn better node embeddings. Attention mechanisms are usually used to assign trainable weights to nodes in neighborhood aggregation, so that the node influence can also participate in the process of neighborhood aggregation. We propose the ATT-MLP model that combines attention mechanism and multi-layer perception(MLP), and applies node attention weights to graph pooling. Experiments on graph prediction show that our algorithm performs better than other baselines on commonly used datasets.

Keywords: Graph Neural Networks; Graph Prediction; Attention Mechanism; Neighborhood Aggregation

1 Introduction

It takes a lot of time and money in drug research. Moreover, screening a large number of compounds is costly. High-throughput screening methods that show molecular activity in drug research from a large number of compounds are popular. By contrast, machine learning is expected to be used to predict the compound activity efficiently using known compound activity information as a training label. In molecular activity prediction, each atom of a molecular is considered as a node of a graph, and a bond is regarded as an edge of a graph. Based on this, neural networks can be used to perform feature extraction[16–18]. Graph convolutional neural network(GCN), which realizes the convolution on graph, is also used to complete such works naturally. Neural graph fingerprints (NGF) [16], GCN by Han et al [17]. and weave module [18] are often used for feature extraction of graph [4].

1.1 Graph Neural Networks

Graph is a kind of non-Euclidean data, which can represent nodes and the relationships (edges) between them [1]. Different from Euclidean data such as text

and image, graph-structured data is irregular, so convolution operation cannot be used in graph datas directly.

In recent years, many researchers have used graph neural networks (GNNs) to process graph structure datas for graph classification [2, 3, 5, 14]. GNNs have important applications in various domains, e.g, chemistry [7], bioinformatics [4, 5], social networks [8], and recommendation systems [9]. GNNs can iteratively update embeddings of nodes to generate a higher-level node representation, and map embeddings of nodes into a graph representation vector combining with readout operation to achieve end-to-end training [6].

The semi-supervised classification model in GCN applies the spectrum convolution classification method to node classification tasks [10]. After adding readout layer, the model can be applied to the graph classification tasks, the model will overfit if there are too many layers in the network. To solve this problem, some researchers use residual networks to construct deep graph neural networks, which can obtain good prediction accuracy, but the time complexity is high [11]. The spatial-convolution method of GCN uses multi-layer neighborhood aggregation to enhance the representation ability of node embeddings, which obtains more efficient graph representation vectors than the spectral convolution model after the readout layer. GIN proposes to use MLP to build the model and batchnorm layer to reduce the over-fitting problem. However, GIN does not consider the hidden information between nodes [8].

1.2 Attention Mechanism

Attention mechanism focuses on the most relevant part to make decisions and obtain more effective node embeddings by aggregating the effective information of neighborhood nodes. Attention mechanism can also process the input information of different scales and focus on the most important part of the input information. Therefore, it also has a good application in graph prediction. Velikovi proposed a graph attention network GAT, which use self-attentional layer to several key challenges of spectral-based graph neural networks simultaneously, and make model readily applicable to inductive as well as transductive problems [12]. DAGCN automatically learns the importance of neighbors at different hops using a novel attention graph convolution layer, and then employs a second attention component, a self-attention pooling layer, to generalize the graph representation from the various aspects of a matrix graph embedding [13]. Zhang propose cardinality preserved attention (GCT-GC) models that can be applied to any kind of attention mechanisms [15].

1.3 Motivation and Contribution

In this work, we add the attention mechanism into the process of hierarchical iteration to enhance the ability of mining node hidden information. We combine attention weights with the graph pooling process to reinforce the expression ability of the graph representation vector.

Contributions of this paper:

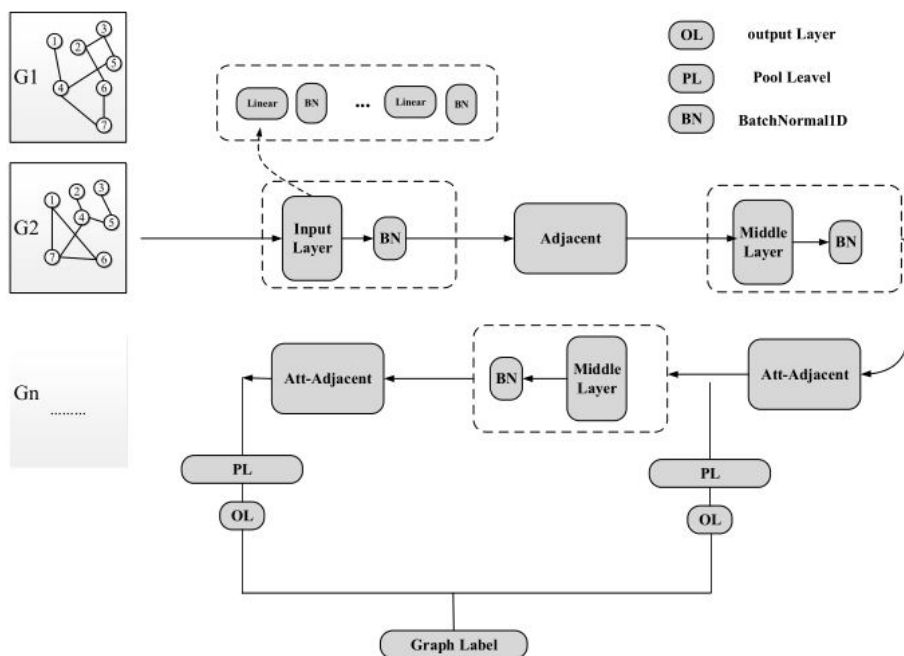


Fig. 1. The architecture of prediction model.

(1) A prediction algorithm is proposed to classify molecular graphs. An attention mechanism is added into the process of the neighborhood aggregation to obtain a better prediction performance.

(2) The attention adjacency matrix is combined with graph pooling to obtain a more representative graph representation vector.

(3) Experiment results demonstrate that the prediction accuracy of our algorithm is significantly improved without a significant increase in time complexity.

1.4 Organization

The rest of the paper is organized as follows: Section 2 describes the problem formulation. The experiments and analyses are provided in Section 3. Finally, Section 4 concludes the paper with some discussions and promising future work.

2 METHOD

2.1 The overall framework

We propose the ATT-MLP model by adding the attention mechanism to the process of hierarchical iteration and readout. As shown in Fig.1, the ATT-MLP model mainly includes three parts: the input layer, the middle layer, and the graph pooling layer.

Graph datas are input to the input layer. The attention mechanism needs enough rich node features, so the neighborhood matrix is used to aggregate node features after the input layer to obtain richer node features. The middle layer is MLP layers, a batchnormal layer after the MLP layers prevent over-fitting, the middle layer uses the attention adjacency matrix to aggregate node features, and all attention layers share parameters. Finally, we weight node embeddings by the attention adjacency matrix to obtain the graph representation vector in the graph pooling layer.

$$H^{(l+1)} = MLP^{(l+1)}(I + ATT)H^l \quad (1)$$

As shown in Eq.(1), ATT represents the attention adjacency matrix, which is obtained by adding the node attention vector. Input characteristic matrix $H^{(0)} = \{h_1^{(0)}, h_2^{(0)}, \dots, h_N^{(0)}\}$, Where n represents the number of nodes, $h_v \in R^F$, h_v is the node representation vector, F is the number of node labels. $H^{(l)} = \{h_1^{(l)}, h_2^{(l)}, \dots, h_N^{(l)}\}$ is obtained after the input characteristic matrix passes through the l -th layer MLP .

Node update formulas of ATT-MLP algorithm are Eq.(2) and Eq.(3). $att(h_v, h_u)$ is the attention coefficients computed by adding the node attention vector to measure the relation between v and u , a is the attention vector, W is the attention layer parameter, h_v, h_u are vector embeddings with edge connections, \parallel is the combination two vectors of Wh_v and Wh_u , h_v^l is the vector embedding of v in the l -th layer, $\mu(v)$ is the neighborhood node set of node v .

$$h_v^l = MLP^{(l)}(h_v^{l-1} + \sum_{u \in \mu(v)} att(h_v, h_u)h_u^{(l-1)}) \quad (2)$$

$$att(h_v, h_u) = \frac{\exp(LeakyReLU(a^T[Wh_v \parallel Wh_u]))}{\sum_{k \in \mu(v)} \exp(LeakyReLU(a^T[Wh_v \parallel Wh_k]))} \quad (3)$$

3 EXPERIMENT

To verify the effectiveness of the proposed ATT-MLP algorithm, we compared it with six other algorithms. All algorithms are implemented on Windows operating system and are built on AMD Ryzen5 2600 and NVIDIA GeForce GTX 1070. We use 6 Common public datasets, as shown in Table 1: 4 bioinformatics datasets (MUTAG, PTC, PROTEINS, ENZYMES), 2 movie collaboration datasets (IMDB-B and IMDB-M), - means that the dataset has no node labels, which is uniformly set to 1 in experiments.

Table 1. Graph Classification Datasets

Dataset	Graph number	Classes	Node Labels
MUTAG	188	2	7
PTC	344	2	19
PROTEINS	1112	2	4
ENZYMES	600	6	3
IMDB-B	1000	2	-
IMDB-M	1500	3	-

3.1 Experimental setting

We compare our model(ATT-MLP algorithm) with several baselines: GCN [10], DGCNN [14], GAT [12], DAGCN [13], GIN [8], GCT-GC [15]. We set 3 layers of MLP and 32 hidden layers in ATT-MLP algorithm. We conduct 10-fold cross-validation and repeated the experiments for 10 times independently to obtain the mean value. Each model runs 200 epochs for each dataset. All models are trained with the Adam optimizer and the learning rate is dropped by a factor of 0.8 every 50 epochs with an initial learning rate of 0.01, dropout $\in \{0, 0.5\}$, batchsize $\in \{16, 64\}$ for all datasets, we use the same hyperparametric configuration in all models to get a fair conclusion.

3.2 Experimental Result

Table 2. Classification accuracy on six datasets

Algorithm/Datasets	MUTAG	PTC	PROTEINS	ENZYMES	IMDB-B	IMDB-M
GCN	84.2±5.9	59.1±6.0	74.1±3.1	28.8±6.4	69.3±4.4	49.7±1.0
DGCNN	85.8±7.9	58.6±6.4	75.3±2.4	49.8±9.3	70.0±3.8	47.8±2.9
GAT	86.4±5.7	59.7±6.7	74.7±3.0	30.2±5.6	70.5±4.7	43.5±3.5
DAGCN	86.1±6.1	61.8±9.6	73.7±4.3	40.1±8.8	73.3±2.7	48.3±3.4
GIN	88.1±7.6	64.6±7.0	75.2±2.8	32.5±6.5	73.6±5.3	49.8±3.2
GCT-GC	90.4±4.7	64.0±7.3	76.9±5.1	59.0±3.8	67.8±4.0	45.6±4.3
ATT-MLP	95.6±4.8	74.1±4.7	78.5±4.1	55.2±5.7	75.6±4.6	51.5±2.5

Graph classification results are shown in Table 2. We highlight the best result of each dataset. We observe that ATT-MLP algorithm performs better than simple attention mechanisms and hierarchical methods. ATT-MLP algorithm significantly improves the classification accuracy on most datasets, attention mechanism redistributes node relationships according to node features, the more node labels in the datasets, the greater improvement of ATT-MLP algorithm, which is also the reason why the improvement effect is not obvious on the ENZYMES dataset.

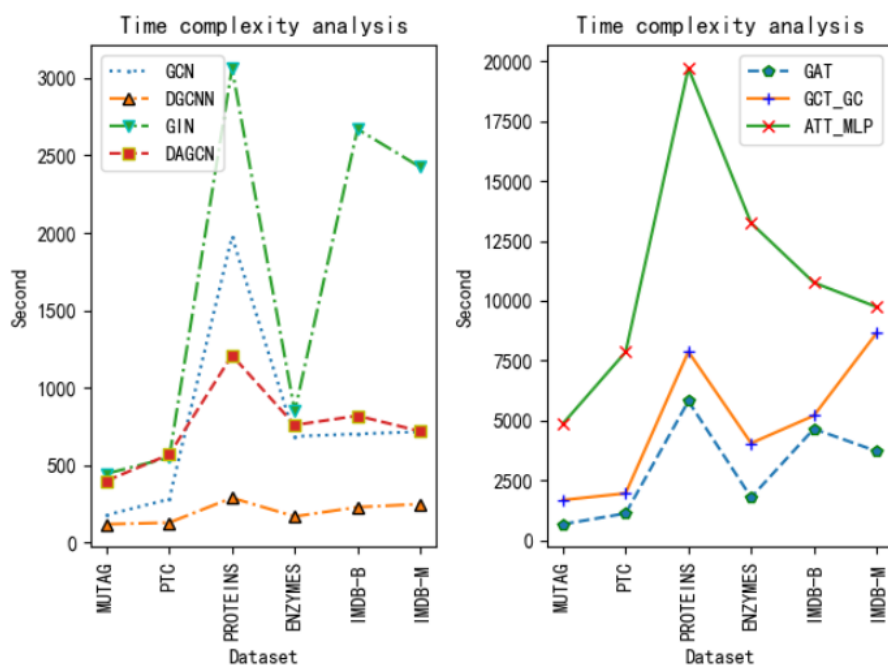


Fig. 2. Time complexity of different algorithms

As shown in Fig.2, the line chart represents the running time of each algorithm on different datasets. ATT-MLP algorithm uses the attention mechanism in neighborhood aggregation. Moreover, the weighted sum of attention matrix is used for node features in graph pooling, so the time complexity of ATT-MLP algorithm is the highest.

4 Conclusion

We propose the method ATT-MLP which embedding the nodes by changing the way of neighborhood aggregation through an attention mechanism, and combine the attention adjacency matrix with graph pooling. ATT-MLP algorithm can re-learn the attention adjacency matrix between nodes based on the hierarchical model. Moreover, our proposed algorithm can mine the hidden information of nodes, and get more representative graph vector representation. Compared to other methods, ATT-MLP achieves competitive performance. However, this model has high time complexity. Learning attention weights more effectively and model parallelization to reduce time complexity are important research tasks in our next work.

References

1. P. Yanardag, S. Vishwanathan, Deep graph kernels, in: KDD '15: Proceedings of the Twenty-First ACM SIGKDD International Conference on Knowledge., 2015, pp. 1365–1374 .
2. Y. Ma, S. Wang, Graph convolutional networks with eigenpooling, in: KDD 19: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining., 2019, pp. 723–731.
3. S. Li, L. Zhang, MutagenPred-GCNs: a graph convolutional neural network-based classification model for mutagenicity prediction with data-driven molecular fingerprints, *Interdiscip. Sci. Comput. Life Sci.* (2021).
4. M. Ohue, R. Ii, Molecular activity prediction using graph convolutional deep neural network considering distance on a molecular graph, in: Proceedings of the Twenty-First ACM SIGKDD International Conference on Knowledge., 2019.
5. Ying Z, You J, Morris C, et al. Hierarchical graph representation learning with differentiable pooling[J]. *Advances in neural information processing systems*, 2018, 31.
6. Zhang M, Cui Z, Neumann M, et al. An end-to-end deep learning architecture for graph classification[C]//Thirty-second AAAI conference on artificial intelligence. 2018.
7. Gilmer J, Schoenholz S S, Riley P F, et al. Neural message passing for quantum chemistry[C]//International conference on machine learning. PMLR, 2017: 1263-1272.
8. Xu K, Hu W, Leskovec J, et al. How powerful are graph neural networks?[J]. *arXiv preprint arXiv:1810.00826*, 2018.
9. Gao C, Yao Q, Jin D, et al. Efficient Data-specific Model Search for Collaborative Filtering[C]//Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 2021: 415-425.

10. Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[J]. arXiv preprint arXiv:1609.02907, 2016.
11. Chen L, Wu L, Hong R, et al. Revisiting graph based collaborative filtering: A linear residual graph convolutional network approach[C]//Proceedings of the AAAI conference on artificial intelligence. 2020, 34(01): 27-34.
12. Veličković P, Cucurull G, Casanova A, et al. Graph attention networks[J]. arXiv preprint arXiv:1710.10903, 2017.
13. Chen F, Pan S, Jiang J, et al. DAGCN: dual attention graph convolutional networks[C]//2019 International Joint Conference on Neural Networks (IJCNN). IEEE, 2019: 1-8.
14. Phan A V, Le Nguyen M, Nguyen Y L H, et al. Dgcnn: A convolutional neural network over large-scale labeled graphs[J]. Neural Networks, 2018, 108: 533-543.
15. Zhang S, Xie L. Improving Attention Mechanism in Graph Neural Networks via Cardinality Preservation[J]. 2019.
16. Duvenaud D, Maclaurin D, Aguilera-Iparraguirre J, et al. Convolutional Networks on Graphs for Learning Molecular Fingerprints[C]// International Conference on Neural Information Processing Systems. MIT Press, 2015.
17. Han A T, Ramsundar B, Pappu A S, et al. Low Data Drug Discovery with One-Shot Learning[J]. ACS Central Science, 2016, 3(4).
18. Kearnes, Steven, McCloskey, et al. Molecular graph convolutions: moving beyond fingerprints.[J]. Journal of Computer-Aided Molecular Design, 2016.