PAY ATTENTION TO MULTI-CHANNEL FOR IMPROVING GRAPH NEURAL NETWORKS

ChungYi Lin[1,2], Shen-Lung Tung[2], Winston H. Hsu[1]
[1]National Taiwan University, [2]Chunghwa Telecom Laboratories

ABSTRACT

We propose Multi-channel Graph Attention (MGAT) to efficiently handle channel-specific representations encoded by convolutional kernels, enhancing the incorporation of attention with graph convolutional network (GCN)-based architectures. Our experiments demonstrate the effectiveness of integrating our proposed MGAT with various spatial-temporal GCN models for improving prediction performance.

1 INTRODUCTION

Recently, GCN-based methods Wu et al. (2019; 2020); Lin et al. (2021) have demonstrated promising results in spatial-temporal prediction. However, these methods assign equal weights to neighboring nodes, leading to suboptimal performance Brody et al. (2022). Therefore, it is intuitive to integrate the Graph Attention Network (GAT) Veličković et al. (2018) into GCN-based methods to dynamically learn weights between nodes.

Nevertheless, GCN-based models contain multiple kernels, each convolving over the input data to create a 'feature map', resulting in a multi-channel representation. Each feature map, referred to as a 'channel', encompasses distinct patterns extracted by different kernels, such as one for sudden traffic congestion and another for cyclical rush hours. GAT assigns equal attention to neighbors without considering channel-specific information, which is critical because different channels may have varying degrees of importance or contributions to the prediction task.

Motivated by this challenge, we propose MGAT, an improved GAT that efficiently computes weights within different channel-specific representations encoded by convolution-based kernels. MGAT learns the spatial-temporal semantics for each channel in parallel. Our experiments demonstrate the effectiveness of our proposed MGAT in conjunction with various ST-GCN models, showing improved prediction performance on two real traffic datasets.

2 MULTI-CHANNEL GRAPH ATTENTION

Preliminaries. H is a multi-channel representation of size $[C \times Z \times D]$ encoded by a 1D CNN kernel as in Wu et al. (2019), where C is the number of channels, Z is the number of attribute nodes (e.g., *time steps* or *spatial locations*), and D is the feature dimension. $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ represents the graph between nodes, with $\mathcal{V} = \{1, \ldots, Z\}$ and $(i, j) \in \mathcal{E}$ indicating an edge from node j to i.

Mechanism of MGAT. To determine the channel-specific weights, we use C independent GATs to explore the correlations between the attribute nodes of each channel's representation. For the c-th channel representation, $H^c = \{h_1^c, h_2^c, \ldots, h_Z^c\} \in \mathbb{R}^{Z \times D}$, we use the *attention coefficient* in GAT, $e(h_i^c, h_j^c)$, to dynamically determine the importance of node j to node i. Notably, we modified the coefficient from previous works Veličković et al. (2018); Brody et al. (2022) by removing the weight matrix W to prevent interference with the weights of the CNN kernel.

Then, these coefficients are normalized across all neighbors of node i based on $\mathcal{G}$, defined as the *attention function*: $\alpha_{ij}^c = softmax(e(h_i^c, h_j^c))$. Next, we compute a weighted sum of the features of node i and its neighbors, and concatenate the results of C independent attention mechanisms, denoted as: $\hat{H}_i = \|_{c=1}^{C}(\sigma(\sum_{j \in N_i} \alpha_{ij}^c h_j^c))$, where $i \in \{1, 2, \ldots, Z\}$, and $\sigma(\cdot)$ is a nonlinear function.

Finally, we denote the function representing the concatenation of the aggregated representations as:

$$MGAT(H, \mathcal{G}) \coloneqq \{\hat{H}_1, \hat{H}_2, ..., \hat{H}_Z\}. \tag{1}$$

Building MGAT-Module. Directly applying the MGAT mechanism to a representation with a large number of channels might result in significant memory usage (e.g., 32 channels used in Wu et al. (2020)). To address this, we adopt 1×1 convolution layers as the Encoder and Decoder components, inspired by the sandwich structure Yu et al. (2018). Given a multi-channel representation H with size $[C \times Z \times D]$, the Encoder downscales the number of channels to $C^{'}$, where $C^{'}$ is smaller than C, and executes the MGAT mechanism. Then, the Decoder upscales the number of channels back to the original size. Residual connections link the Encoder and Decoder to stabilize the training. Formally, the output of the MGAT-Module, denoted as $\hat{H}$, is given by:

$$\hat{H} = Decoder(MGAT(Encoder(H), \mathcal{G})) + H. \tag{2}$$

Integrating MGAT-Module with target ST-GCN. We replace temporal and spatial modeling in existing ST-GCNs with the MGAT-Module, and Z can represent either historical time steps or road sensors in different views. The MGAT-Module cooperates with $\mathcal{G}$ proposed in the target ST-GCNs.

3 EXPERIMENTS

Datasets and Baselines. We use the PEMS08-flow and PEMS08-speed datasets from Guo et al. (2019), which contain traffic flow and speed data, respectively. These datasets were collected from 170 road sensors during July-August 2016 and recorded every 5 minutes. For baselines, we selected three open-source ST-GCN models: Graph WaveNet (GWNET) Wu et al. (2019), MTGNN Wu et al. (2020), and MPNet Lin et al. (2021). Other experimental setups are given in Appendix A.

Results and Analysis. Table 1 summarizes the evaluation of integrating the MGAT-Module with state-of-the-art ST-GCN models from short- to long-term predictions on two datasets. We observe that: (1) The MGAT-Module improved prediction performance for all prediction steps, demonstrating its effectiveness in extracting correlations for various ST-GCN models. (2) ST-GCN models integrated with MGAT-Module consistently outperformed the original ones, indicating the strong generality of our approach. (3) The ST-GCNs with the MGAT-Module were more effective for long-term prediction tasks, as evidenced by the increasing performance gaps compared to vanilla ST-GCN models as the prediction steps increased.

Table 1: The MAE results of ST-GCN models with or without MGAT-Module on two datasets.

Steps	GWNET	GWNET*	MTGNN	MTGNN*	MPNet	MPNet*
Flow						
3	13.64 ± 0.11	**13.42** ± 0.11	14.04 ± 0.03	**13.65** ± 0.01	13.89 ± 0.18	**13.50** ± 0.04
6	14.65 ± 0.12	**14.27** ± 0.13	15.04 ± 0.01	**14.54** ± 0.02	15.01 ± 0.31	**14.36** ± 0.03
9	15.47 ± 0.15	**14.98** ± 0.16	15.94 ± 0.06	**15.25** ± 0.05	15.89 ± 0.18	**15.10** ± 0.04
12	16.30 ± 0.16	**15.69** ± 0.13	17.01 ± 0.11	**16.02** ± 0.07	16.79 ± 0.55	**15.83** ± 0.02
Avg.	14.67 ± 0.13	**14.30** ± 0.11	15.11 ± 0.01	**14.55** ± 0.02	15.01 ± 0.13	**14.41** ± 0.02
Speed						
3	1.11 ± 0.004	**1.07** ± 0.005	1.13 ± 0.005	**1.11** ± 0.004	1.10 ± 0.004	**1.08** ± 0.003
6	1.38 ± 0.004	**1.30** ± 0.012	1.40 ± 0.006	**1.35** ± 0.003	1.39 ± 0.005	**1.33** ± 0.002
9	1.53 ± 0.005	**1.43** ± 0.016	1.55 ± 0.006	**1.47** ± 0.005	1.57 ± 0.010	**1.46** ± 0.002
12	1.63 ± 0.008	**1.53** ± 0.017	1.67 ± 0.007	**1.58** ± 0.002	1.69 ± 0.011	**1.57** ± 0.001
Avg.	1.33 ± 0.003	**1.26** ± 0.011	1.35 ± 0.005	**1.30** ± 0.002	1.35 ± 0.005	**1.28** ± 0.001

'*' indicates the model with the proposed MGAT-Module.

4 CONCLUSION

The proposed MGAT effectively handles multi-channel representations and its extension module generally improves GCN-based prediction performance. We provide insights into how incorporating attention with multi-channel representations in GCN-based architectures enhances their effectiveness and offers practical implications for real-world applications, such as traffic prediction.

URM STATEMENT

Author One meets the URM criteria of ICLR 2023 Tiny Papers Track

REFERENCES

Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks? In *International Conference on Learning Representations*, 2022.

Shengnan Guo, Youfang Lin, Ning Feng, Chao Song, and Huaiyu Wan. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In *Proc. of AAAI*, pp. 922–929, 2019.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations*, 2018.

Chung-Yi Lin, Hung-Ting Su, Shen-Lung Tung, and Winston H. Hsu. Multivariate and propagation graph attention network for spatial-temporal prediction with outdoor cellular traffic. In *Proceedings of the 30th ACM International Conference on Information Knowledge Management*, 2021.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. Graph wavenet for deep spatial-temporal graph modeling. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, 2019.

Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang. Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery and data mining*, 2020.

Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018.

A APPENDIX: REPRODUCIBILITY

Settings and Hyperparameters. We used a 70-20-10 train-test-validation split following Li et al. (2018) and trained each model for 100 epochs with 10 repeats. The samples used a 12-time step historical time window, and the prediction horizon ranged from 1 to 12 steps (i.e., 5 to 60 minutes). We employed curriculum learning Wu et al. (2020) during training to gradually increase the prediction length. The Encoder and Decoder of the MGAT-Module had 8 ($C^{'}$) and 32 (C) output channels.