2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC)
October 6-10, 2024. Sarawak, Malaysia

Temporal MLP Bridges the Gap between Embedding and Attention for Multivariate Time Series Forecasting

Zhinan Xie[1], Qi Zheng[2]✉, and Yaying Zhang[2]

Abstract— **Multivariate time series forecasting is crucial across various applications. In recent years, numerous studies adopt embedding layer and Attention mechanism to extract the intricate spatio-temporal features of time series. This involves directly transmitting the concatenated embeddings into the Attention mechanism. However, they generally overlook the importance of sending the integrated information in the embeddings into the Attention mechanism in a more appropriate way. To address this, we propose an intuitive network model with Temporal MLP Bridging the gap between Embedding and Attention (TMBEA) to deal with the above issue. Specifically, we explore a light-weight bridge with simple Multi-Layer Perceptrons (MLPs) fusing features along the temporal dimension, processing the embeddings before feeding them into the canonical Attention networks, which help embeddings to better align with the subsequent Attention networks. Experiments on real-world datasets, traffic datasets and air pollutant concentration datasets, demonstrate the efficiency of model. Further studies also show the capacity of bridge in improving the robustness of the model.**

I. Introduction

Accurate multivariate time series forecasting is crucial for fields like weather, electricity, and traffic [1]–[3] due to its inherent temporal and spatial dependencies.

Over the years, significant advancements have been made in this field. Initially, deep learning models like Recurrent Neural Networks (RNNs) were used to analyze temporal features [4]–[6]. Later, Graph Convolutional Networks (GCNs) became prominent for spatial graphs, often integrated with RNNs [7], [8], Graph Neural Networks (GNNs) [9], [10], and Temporal Convolutional Networks (TCNs) [2], [11], [12] to enhance predictive performance. Furthermore, to better capture intricate spatio-temporal dependencies, Attention mechanisms have become increasingly prevalent. Efforts in Attention-based models focus on: **(1) Components of embeddings to cooperate with Attention mechanism**: GMAN [13] and PDFormer [14] utilized spatial embedding by Node2Vec [15] and Spatial Graph Laplacian Embedding respectively to emphasize the spatial feature. **(2) The combination of Attention mechanism with other models**: To formulate a graph that can represent the spatio-temporal correlations more accurately, some studies [16], [17] have delved into the fusion of GCN and Convolutional Long Short-Term Memory (ConvLSTM) architectures with Attention mechanisms. **(3) The modification on the original Attention structure** [18]–[20]: These studies aim to mitigate the computational resource demands of attention models, also preserving their capability in time series forecasting.

While each contributes valuable insights to Attention-based research, they have not fully addressed direct information transfer between independent modules without processing. Notably, there is an oversight regarding the **bridge** connecting embeddings with subsequent Attention architectures, which enhances the handling of embeddings to better align with Attention networks.

Inspired by this, in this paper, we look deep into the bridge. From recent studies, we derive the following observations regarding the integration of embedding and Attention: **(1) Embedding without Attention**: STID [21] concatenates spatial and temporal embeddings, processing them with MLP networks, achieving remarkable performance. This shows that even simple networks can make embeddings effective, motivating further embedding processing for better synergy with Attention mechanisms. **(2) Embedding with Attention**: STAEformer [22] adopts an adaptive embedding, which makes the vanilla Transformer [23] perform better. However, it lacks adaptation to refine the embeddings before feeding them into the Transformer architecture. **(3) Bridge with Attention** [20]: utilizing causal convolution as a bridge to generate queries and keys for Attention has proven effective for capturing local information for long-term prediction. However, causal convolution, limited to preceding text, misses global information integration, leading to potential information asymmetry. This motivates building a bridge across the temporal dimension to capture full contextual features for queries, keys, and values, which are crucial components for Attention mechanisms.

To achieve the above purpose, we proposed a straightforward yet effective MLP network playing as bridge to fuse temporal features within the embeddings, thereby attaining a comprehensive global receptive field. Subsequently, the refined embeddings play as query, key and value in temporal Attention, followed by spatial Attention and regression layer. This concise architecture has achieved favorable outcomes across five datasets. The experiments meanwhile demonstrate that the bridge enhances the robustness of the model.

II. Problem Definition

Given historical multivariate time series $[X_{t-H+1}, \cdots, X_t]$ with H previous time slots, we aim to learn a function f which is capable of predicting

*This work is partly supported by the National Natural Science Foundation of China under Grant 72342026.

[1]Zhinan Xie is with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China xiezhn@shanghaitech.edu.cn

[2]Qi Zheng and Yaying Zhang are with the Department of Computer Science, Tongji University, Shanghai, China {zhengqi97, yaying.zhang}@tongji.edu.cn

2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC) | 978-1-6654-1020-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/SMC54092.2024.10831557

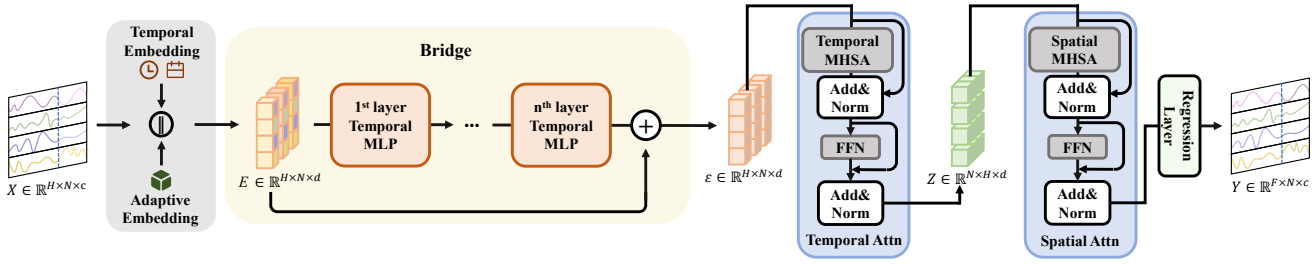


Fig. 1: The overall framework of our proposed model.

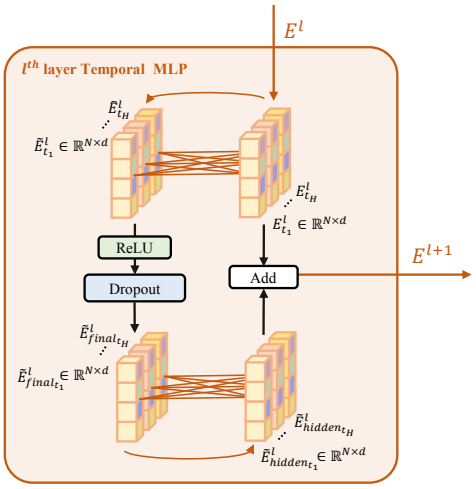


Fig. 2: The architecture of a single layer of the temporal MLP

the future multivariate time series $[Y_{t+1}, \cdots, Y_{t+F}]$ with F time slots.

$$[X_{t-H+1}, \cdots, X_t] \xrightarrow{f} [Y_{t+1}, \cdots, Y_{t+F}] \tag{1}$$

$X_i, Y_i \in \mathbb{R}^{N \times c}$ where N represents the number of spatial nodes and c denotes the dimension of the feature space.

III. Methodology

The overall framework of our proposed model is as shown in Fig. 1. It mainly has five steps, which are embedding layer, Temporal MLP (the bridge), Multi-head Temporal Attention, Multi-head Spatial Attention, and Regression layer.

A. Embedding Layer

In order to provide more informative feature representation on spatial and temporal for subsequent learning process, embedding layer is employed to project the input variables into an alternative space.

Apart from the historical features $X \in \mathbb{R}^{H \times N \times c}$ (e.g. the traffic flow in traffic datasets), we derive two time information: the time-of-day information $X_{tod} \in \mathbb{R}^{H \times N \times 1}$ and day-of-week information $X_{dow} \in \mathbb{R}^{H \times N \times 1}$. Since there are 7 days a week and predefined T time frames a day, X_{dow} only has 7 possible values, and T values are available for X_{tod}. Thus, we initialize the embedding dictionary for each kind of time information as $I_{dow} \in \mathbb{R}^{7 \times d_2}$ for day-of-week and $I_{tod} \in \mathbb{R}^{T \times d_3}$ for time-of-day (d_2 and d_3 are the dimensions of the embedding space). When we need to embed the time information for each node in the historical time steps, we just need to pick the corresponding slices in I_{dow} and I_{tod}, denoting the embedded nodes as $E_{tod} \in \mathbb{R}^{H \times N \times d_2}$ and $E_{dow} \in \mathbb{R}^{H \times N \times d_3}$.

First, we project all features of input into a hidden space dimension, represented as $E_w \in \mathbb{R}^{H \times N \times d_1}$.

$$E_w = Linear(X \| X_{tod} \| X_{dow}) \tag{2}$$

Next, we apply learnable temporal embeddings $E_{tod} \in \mathbb{R}^{H \times N \times d_2}$ and $E_{dow} \in \mathbb{R}^{H \times N \times d_3}$ to characterize temporal information. Additionally, adaptive embedding has been proved to be an efficient structure for Transformers [22], so we adopt a randomly initialized adaptive embedding $E_a \in \mathbb{R}^{H \times N \times d_4}$ to capture intricate spatio-temporal relation.

By concatenating all the embeddings above along the feature dimension, we obtain the embedding $E \in \mathbb{R}^{H \times N \times d}$ indicating both feature and time information, where $d = d_1 + d_2 + d_3 + d_4$ and $\|$ denotes the concatenation along the feature dimension.

$$E = E_w \| E_{tod} \| E_{dow} \| E_a \tag{3}$$

However, employing embeddings directly in this manner may result in overly discrete information, consequently diminishing the efficiency of the Attention layers. Hence, we propose a solution to this issue in the subsequent part, which is an approach to fuse embeddings.

B. Temporal MLP

We propose an n-layer MLP network operating along the temporal dimension of the embeddings to refine features of each time phase, thereby incorporating global contextual temporal information into the embedding at each time step. This module maintains the shape of the embedding unchanged.

As shown in Fig. 2, for the l_{th} layer of Temporal MLP, embedding E^l (when $l = 1$, E^l is E generated in Eq. (3)) can be partitioned along the temporal dimension into H vectors, yielding $E^l_{t_1}, \cdots, E^l_{t_H} \in \mathbb{R}^{N \times d}$. We aim for embeddings of each temporal phase to encompass a fraction of information

from other temporal phases, thus we propose the fusion on global contextual information in the form of Eq. (4).

$$\widetilde{E}^l_{t_i} = w^l_{i1}E^l_{t_1} + \cdots + w^l_{iH}E^l_{t_H}, i = 1, \cdots, H \tag{4}$$

where w_{ij} are learnable parameters. This approach characterizes time continuously rather than discretely, providing useful background for the subsequent attention mechanism. For short time series prediction, it is efficient, demanding only $H \times H$ parameters, compared to $N \times N$ parameters for each layer in the spatial dimension. When the number of spatial nodes exceeds the number of historical time steps, this configuration enhances efficiency and reduces memory usage.

Following the global fusion of embeddings across the temporal dimension, we proceed to apply activation and regularization function to the output:

$$\widetilde{E}^l_{hidden} = Dropout(\sigma(\widetilde{E}^l)) \tag{5}$$

where σ serves as an activation function, being set to ReLU function, while $Dropout$ function as the regularization mechanism. Following, we repeat Eq. (4) again towards $\widetilde{E}^l_{hidden} \in \mathbb{R}^{H\times N\times d}$ gaining $\widetilde{E}^l_{final} \in \mathbb{R}^{H\times N\times d}$. Afterward, we utilize a residual layer to prevent degeneration during learning as shown in Eq. (6).

$$E^l_{final} = \widetilde{E}^l_{final} + E^l \tag{6}$$

For the next layer, we employ the output of the preceding layer E^l_{final} as the input E^{l+1}, and stack another block with the same aforementioned process (Eq. (4), Eq. (5), Eq. (6)), aiming to attain a more refined fusion effect. After n layers, we obtain an overall output denoted as E^n_{final}.

Finally, we adopt residual connection again to underscore the original embedding information E generated in Eq. (3), and obtain the $\mathcal{E} \in \mathbb{R}^{H\times N\times d}$:

$$\mathcal{E} = E^n_{final} + E \tag{7}$$

$\mathcal{E}$ becomes the source for query, key and value in the following temporal multi-head self-Attention layer. Simultaneously, the temporal MLP module preserves the shape of the embeddings at each layer to retain more realistic information.

C. Attention Mechanism and Regression Layer

Given the intrinsic relationships among information from different time phases at different spatial nodes, we aim to utilize attention mechanisms across temporal and spatial dimensions to effectively capture these correlations.

Adopting the canonical Transformer model architecture [24], we first apply the Attention mechanism along the temporal dimension. We transpose $\mathcal{E} \in \mathbb{R}^{H\times N\times d}$ into temporal embeddings $\mathcal{E}t_1, \cdots, \mathcal{E}t_H \in \mathbb{R}^{N\times d}$. Utilizing fully connected layers, $FC(\cdot)$, we derive the query, key, and value matrices $Q_{tmp}, K_{tmp}, V_{tmp} \in \mathbb{R}^{N\times H\times d}$ as follows:

$$\begin{aligned} Q_{tmp} &= FC_q(reshape(\mathcal{E}_{t_1} \| \cdots \| \mathcal{E}_{t_H})) \\ K_{tmp} &= FC_k(reshape(\mathcal{E}_{t_1} \| \cdots \| \mathcal{E}_{t_H})) \\ V_{tmp} &= FC_v(reshape(\mathcal{E}_{t_1} \| \cdots \| \mathcal{E}_{t_H})) \end{aligned} \tag{8}$$

TABLE I: Details for Datasets

Dataset	#Sensors	#Timesteps	TimeRange
PeMS04	307	16992	01/2018-02/2018
PeMS08	170	17856	07/2016-08/2016
AQI	35	17382	01/2015-12/2016
PM10	35	17382	01/2015-12/2016
PM2.5	35	17383	01/2015-12/2016

Note that the $FC(\cdot)$ operation denotes the linear projection along the feature dimension d. $Q_{tmp}, K_{tmp}, V_{tmp}$ are then sent into Multi-head Self-Attention $MHSA(\cdot)$ and feed forward networks $FFN(\cdot)$:

$$\begin{aligned} Z_{Att} &= LayerNorm(MHSA(Q_{tmp}, K_{tmp}, V_{tmp}) + \mathcal{E}) \\ Z &= LayerNorm(FFN(Z_{Att}) + Z_{Att}) \end{aligned} \tag{9}$$

where $Z \in \mathbb{R}^{N\times H\times d}$ is the output.

And then, we extend the Attention mechanism to the spatial dimension, involving partitioning of Z into $Z_{n_1}, \cdots, Z_{n_N} \in \mathbb{R}^{H\times d}$ to generate $Q_{spt}, K_{spt}, V_{spt} \in \mathbb{R}^{H\times N\times d}$. Repeat the above Add and Norm step (Eq. (9)), resulting in an output of $Y_0 \in \mathbb{R}^{H\times N\times d}$.

For the regression layer, it operates on the output $Y_0 \in \mathbb{R}^{H\times N\times d}$ derived from the feature enhanced by spatial attention. Initially, we transform Y_0 into $Y_1 \in \mathbb{R}^{N\times F\cdot d}$, followed by prediction utilizing Eq. (10).

$$Y = reshape(FC_{regression}(Y_1)) \tag{10}$$

Consequently, we obtain the predicted future series $Y \in \mathbb{R}^{F\times N\times c}$.

IV. Experiments

To verify the effectiveness of TMBEA, experiments are conducted to compare it with baseline methods.

A. Experimental Setup

Datasets: We evaluate our proposed structure on five real-world datasets in total. (1) Two traffic forecasting benchmarks, i.e. PeMS04 and PeMS08. The datasets depict the traffic flows spanning the freeway system across all major metropolitan areas of the State of California[1]. (2) Three datasets for air pollutant concentration, namely AQI, PM10 and PM2.5. The datasets[2] cover the period from 2015 to 2016, with air pollutant concentrations in Beijing recorded hourly. Missing values are filled with linear interpolation.

In summary, the time interval for the two traffic datasets is 5 minutes, resulting in $T = 288$ time frames per day. And the time interval for the three air pollutant concentration datasets is 1 hour, leading to $T = 24$ time frames per day. Further details are provided in Table I.

Baselines: We compare TMBEA with the following baselines. (1) HI [25]: The typical traditional method, utilizing the feature values of the last F time steps of the input H time steps as the prediction. (2) GWNet [12], AGCRN

[1] http://pems.dot.ca.gov
[2] https://quotsoft.net/air

TABLE II: Performance on PeMS04 and PeMS08

Dataset	PeMS04			PeMS08		
Metrics	**RMSE**	**MAE**	**MAPE**	**RMSE**	**MAE**	**MAPE**
HI	128.20	94.25	153.70%	120.12	86.04	166.72%
GWNet	30.44	19.01	13.32%	23.69	14.80	9.44%
AGCRN	32.75	20.04	13.47%	24.86	15.66	10.47%
STGCN	31.56	19.66	13.32%	25.77	16.46	11.02%
STID	**29.90**	18.35	12.56%	23.93	14.44	9.63%
STAEformer	30.11	18.23	**11.89%**	23.33	13.58	9.05%
TMBEA	29.92	**18.19**	12.00%	**23.20**	**13.48**	**8.89%**

[7], STGCN [2]: based on graph neural networks to capture spatial dependencies in the adjacency matrix of graph. (3) STID [21]: Explore a concise and efficient model, comprising embedding layer, followed by MLPs. (4) STAEformer [22]: Employing embeddings applicable to Transformers, including adaptive embeddings, together with vanilla Transformer.

Metrics: We test the performance of all baseline models and TMBEA with three widely adopted metrics in multivariate time series forecasting: Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE). We compare the average performance for all the datasets over the horizons. The predicted value of the k^{th} sample is $\hat{y_k}$, and the real value of the k^{th} sample is y_k. m is the number of samples. MAE, RMSE and MAPE can be formulated as:

$$MAE = \frac{1}{m}\sum_{k=1}^{m}|\hat{y_k} - y_k|. \tag{11}$$

$$RMSE = \sqrt{\frac{1}{m}\sum_{k=1}^{m}(\hat{y_k} - y_k)^2}, \tag{12}$$

$$MAPE = \frac{100\%}{m}\sum_{k=1}^{m}|\frac{\hat{y_k} - y_k}{y_k}|. \tag{13}$$

Implementation: We implement the model with Pytorch 1.12.0 on an NVIDIA RTX 3060 GPU. For PeMS04 and PeMS08, we set the number of input historical steps H to 12, and the horizon F to 12. For all the air pollutant concentration datasets, we set two prediction tasks: 1) $H = 8$, $F = 4$. 2) $H = 4$, $F = 1$. All the five datasets are divided into train, validation and test sets with ratio 6:2:2. The feature dimension c is 1, and the temporal MLP layer n is set to 3. We train the models with Adam optimizer with an initial learning rate of 0.001, and set the batch size to 8.

B. Performance Study

Performance on PeMS04 and PeMS08: Table II compares the average performance of TMBEA and baselines across 12 predicted time steps. The best results are in bold, and the second-best are underlined. TMBEA outperforms convolution-based models GWNet, STGCN, and AGCRN, highlighting the efficiency of Attention-based models. Notably, TMBEA surpasses recent models STID and STAEformer on most metrics. Outperforming STID indicates that the Attention mechanism extracts deeper spatio-temporal features, while surpassing STAEformer suggests that the straightforward temporal processing of embeddings effectively enhances Attention mechanism performance.

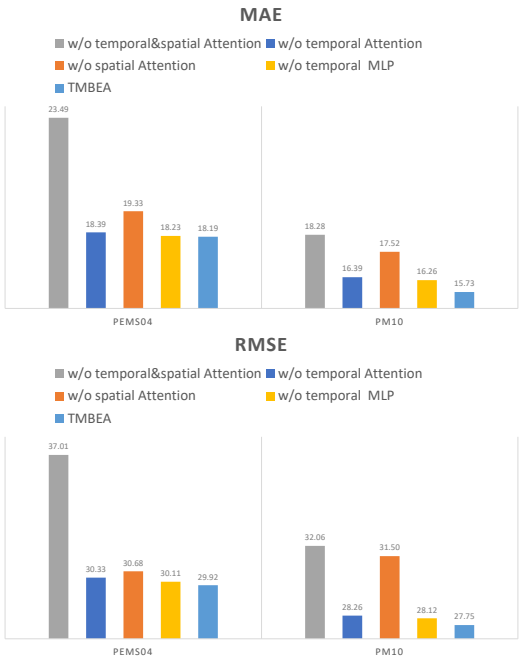


Fig. 3: Results of four ablation studies compared to the original TMBEA on PeMS04 and PM10

Performance on Air Pollutant Concentration Datasets: As depicted in Table III and IV, TMBEA outperforms the baseline model on most metrics, demonstrating its generalization capacity. Notably, TMBEA performs better when predicting 4 future time steps with 8 historical steps compared to predicting 1 step with 4 steps, indicating its superior ability to capture spatio-temporal features for longer-term predictions.

C. Ablation Study

To verify the effectiveness of each part of TMBEA, we conduct the experiments on PeMS04 ($H = 12, F = 12$) and PM10 ($H = 8, F = 4$): 1) **w/o temporal MLP** removes the temporal MLP layer we proposed. 2) **w/o spatial Attention** removes the whole spatial Attention block. 3) **w/o temporal Attention** removes the whole temporal Attention block. 4) **w/o temporal & spatial Attention** removes both the whole temporal and spatial Attention blocks. 5) **spatial MLP + spatial Attention** replaces the original temporal MLP with a network of identical structure as our proposed temporal MLP, while designed to operate spatially. And removes the whole temporal Attention block. This modification results in feature extraction exclusively along the spatial axis. 6) **causal convolution & Attention** replaces the temporal MLP in TMBEA with causal convolution.

We have seen results as below:

- **Results of four ablation studies**
  In Fig. 3, the TMBEA **w/o temporal MLP** experiment shows a performance drop, demonstrating the effective-

TABLE III: Performance on Air Pollutant Concentration Datasets (4 historical steps predicting 1 future step)

Dataset	AQI			PM10			PM2.5		
Metrics	**RMSE**	**MAE**	**MAPE**	**RMSE**	**MAE**	**MAPE**	**RMSE**	**MAE**	**MAPE**
HI	68.82	43.23	74.75%	75.69	50.21	138.15%	69.13	45.15	223.49%
GWNet	**5.40**	1.66	2.40%	18.27	9.62	15.78%	14.91	8.07	**17.80%**
AGCRN	5.41	1.66	2.33%	18.68	10.13	16.21%	15.18	8.32	19.26%
STGCN	10.58	6.64	12.47%	23.27	14.37	27.51%	17.02	9.28	19.39%
STID	5.58	1.62	2.37%	20.12	10.59	16.72%	16.80	9.00	20.84%
STAEformer	5.41	1.60	2.34%	18.24	9.51	15.14%	14.56	7.81	18.42%
TMBEA	5.42	**1.57**	**2.31%**	**18.18**	**9.46**	**14.93%**	**14.29**	**7.71**	18.58%

TABLE IV: Performance on Air Pollutant Concentration Datasets (8 historical steps predicting 4 future steps)

Dataset	AQI			PM10			PM2.5		
Metrics	**RMSE**	**MAE**	**MAPE**	**RMSE**	**MAE**	**MAPE**	**RMSE**	**MAE**	**MAPE**
HI	47.47	69.91	79.71%	81.14	58.68	152.33%	74.61	52.80	243.74%
GWNet	8.21	3.59	4.76%	29.86	16.71	26.57%	28.37	15.62	37.56%
AGCRN	8.47	3.97	5.19%	31.27	18.07	29.63%	30.31	16.91	36.94%
STGCN	8.62	4.27	5.75%	30.33	18.32	28.91%	28.05	15.62	34.59%
STID	9.26	4.01	5.44%	33.89	19.32	32.83%	32.30	17.52	43.54%
STAEformer	8.29	3.68	4.92%	28.12	16.26	26.46%	26.48	14.46	33.64%
TMBEA	**8.09**	**3.49**	**4.69%**	**27.75**	**15.73**	**25.86%**	**25.79**	**14.09**	**32.57%**

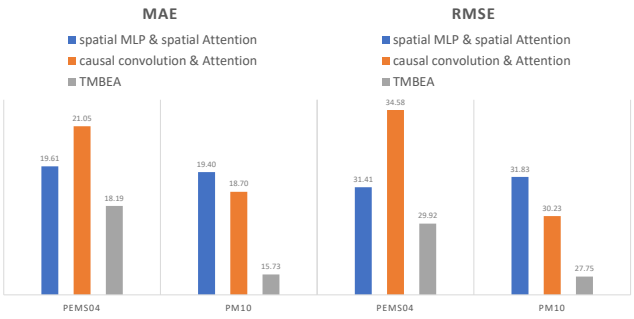


Fig. 4: Results of two variant experiments compared to the original TMBEA on PeMS04 and PM10

ness of our proposed temporal MLP. For both datasets, TMBEA **w/o spatial Attention** and **w/o temporal Attention** also show decreased accuracy, highlighting the importance of both Attention blocks, especially the spatial one. TMBEA **w/o temporal & spatial Attention** performs the worst, lacking the integration of both blocks. This confirms that the Attention mechanism is crucial for capturing intricate spatial and temporal features.

- **Results of two variant experiments**
  As shown in Fig. 4, the model **spatial MLP + spatial Attention** highlights the importance of extracting both temporal and spatial information and the benefit of fusing temporal features within embeddings during the bridge block. The model **causal convolution & Attention** indicates that integrating full contextual information is vital, as causal convolution captures only preceding information.

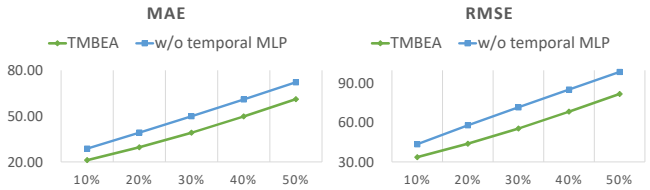
D. Robust Study

To assess the efficacy of our design bridge, we conduct a robust study on the PeMS08 dataset by introducing missing data. We randomly set zeros on the train, test sets, or both to create missing data, with mask rates of 10%, 20%, 30%, 40%, and 50%. Masking the training set simulates defective training data, while masking the test set replicates real-world data imperfections. This allows us to evaluate if the temporal MLP can enhance the robustness of TMBEA under these conditions.

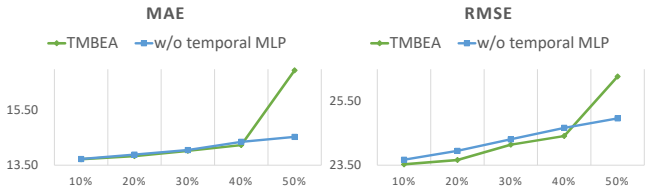
As shown in Fig. 5, we compare TMBEA with TMBEA **w/o temporal MLP** on PeMS08. At lower mask rates, TMBEA outperforms the variant **w/o temporal MLP**. As the mask rate increases, the advantage of TMBEA with temporal MLP becomes more pronounced, demonstrating its ability to maintain predictive accuracy in harsh conditions. This highlights the bridge's unique benefit of handling incomplete data effectively. Despite challenges with severely incomplete training sets, TMBEA offers a reliable and robust real-world solution.

V. Conclusion

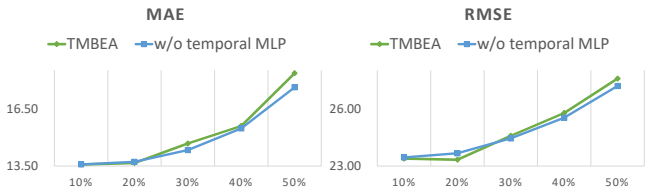
In this paper, we focus on how to deal with embeddings so that the subsequent Attention mechanism can better extract the temporal and spatial features of the time series. We proposed a simple temporal MLP network playing as a **bridge** connecting embedding and Attention mechanism. The model with embedding, bridge, Attention mechanism achieves better performance through two traffic datasets, and three real-life air pollutant concentration datasets. Further studies demonstrate that our proposed bridge can help the structure resist harsh test environment. These results suggest that the bridge can enhance the impact of embeddings,



(a) mask test set only



(b) mask both test and train sets



(c) mask train set only

Fig. 5: the MAE and RMSE of TMBEA and **w/o temporal MLP** on PeMS08 when we train and test on the dataset with mask rate between 10% and 50% on different partitions (mask train set or test set or both)

providing a promising direction for further research into embeddings. However, this paper does not delve deeply into the design of embedding layer, leaving space for future research aiming at creating simpler embedding yet preserving richer spatio-temporal information.

REFERENCES

[1] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," *arXiv preprint arXiv:2211.14730*, 2022.

[2] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," *arXiv preprint arXiv:1709.04875*, 2017.

[3] M. Liu, A. Zeng, M. Chen, Z. Xu, Q. Lai, L. Ma, and Q. Xu, "Scinet: Time series modeling and forecasting with sample convolution and interaction," *Advances in Neural Information Processing Systems*, vol. 35, pp. 5816–5828, 2022.

[4] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[5] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[6] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling long-and short-term temporal patterns with deep neural networks," in *The 41st international ACM SIGIR conference on research & development in information retrieval*, 2018, pp. 95–104.

[7] L. Bai, L. Yao, C. Li, X. Wang, and C. Wang, "Adaptive graph convolutional recurrent network for traffic forecasting," *Advances in neural information processing systems*, vol. 33, pp. 17 804–17 815, 2020.

[8] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," *arXiv preprint arXiv:1707.01926*, 2017.

[9] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE transactions on neural networks*, vol. 20, no. 1, pp. 61–80, 2008.

[10] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, "Connecting the dots: Multivariate time series forecasting with graph neural networks," in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 2020, pp. 753–763.

[11] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[12] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph wavenet for deep spatial-temporal graph modeling," *arXiv preprint arXiv:1906.00121*, 2019.

[13] C. Zheng, X. Fan, C. Wang, and J. Qi, "Gman: A graph multi-attention network for traffic prediction," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 01, 2020, pp. 1234–1241.

[14] J. Jiang, C. Han, W. X. Zhao, and J. Wang, "Pdformer: Propagation delay-aware dynamic long-range transformer for traffic flow prediction," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, no. 4, 2023, pp. 4365–4373.

[15] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.

[16] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 922–929.

[17] Z. Lin, M. Li, Z. Zheng, Y. Cheng, and C. Yuan, "Self-attention convlstm for spatiotemporal prediction," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 11 531–11 538.

[18] S. Liu, H. Yu, C. Liao, J. Li, W. Lin, A. X. Liu, and S. Dustdar, "Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting," in *International conference on learning representations*, 2021.

[19] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, no. 12, 2021, pp. 11 106–11 115.

[20] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X. Wang, and X. Yan, "Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting," *Advances in neural information processing systems*, vol. 32, 2019.

[21] Z. Shao, Z. Zhang, F. Wang, W. Wei, and Y. Xu, "Spatial-temporal identity: A simple yet effective baseline for multivariate time series forecasting," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2022, pp. 4454–4458.

[22] H. Liu, Z. Dong, R. Jiang, J. Deng, J. Deng, Q. Chen, and X. Song, "Spatio-temporal adaptive embedding makes vanilla transformer sota for traffic forecasting," in *Proceedings of the 32nd ACM international conference on information and knowledge management*, 2023, pp. 4125–4129.

[23] R. Al-Rfou, D. Choe, N. Constant, M. Guo, and L. Jones, "Character-level language modeling with deeper self-attention," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 3159–3166.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[25] Y. Cui, J. Xie, and K. Zheng, "Historical inertia: A neglected but powerful baseline for long sequence time-series forecasting," in *Proceedings of the 30th ACM international conference on information & knowledge management*, 2021, pp. 2965–2969.