https://doi.org/10.1007/sxxxxx-yyy-zzzz-1

1 Experimental details and further analytical results

1.1 Real Dataset Details

The Shenzhen dataset is an open-source collection from a mobile application [1], containing charging demand records for 247 urban areas over 30 days, with a sampling interval of 5 minutes. The Hefei dataset is self-collected from Hefei. We collect real-time data from 117,657 EVs over a three-month span from October to December 2023. We extract 4,084,986 charging events based on vehicle charging states and apply DBSCAN clustering with a radius of 50 meters and a minimum density of 100 points to identify charging station locations, resulting in 817 sites. Tab. 1 shows data samples of the Hefei charging dataset. We determine whether a vehicle is charging based on its charging state, where a charging state of 1 indicates that the vehicle is parked and charging. We extract charging events from the time series data. Subsequently, we use the DBSCAN algorithm, as detailed in Algorithm 1, to cluster all the locations where charging events occur to identify the locations of charging stations. The number of charging vehicles within 50 meters of the charging station center is defined as the station's traffic flow. We are working towards anonymizing and publicly releasing the Hefei dataset.

Algorithm 1 Charging Station Identification

Require: EV Charging data X (time, location)
Ensure: Set of Charging Stations C

Split X into half-hourly groups
Initialize empty set C
$R \leftarrow$ Charging station radius
$D_{\min} \leftarrow$ Minimum distance between stations
for each group G in split data **do**
  Perform DBSCAN clustering on locations in G
  for each cluster C_i in clustering result **do**
    $P_i \leftarrow$ Centroid of C_i
    $N_i \leftarrow$ Number of vehicles in C_i
    if $N_i > 1$ **then**
      $P_j \leftarrow$ Closest point in C to P_i
      $N_j \leftarrow$ Number of vehicles at P_j
      if distance$(P_i, P_j) > R$ **then**
        Add (P_i, N_i) to C
      else
        $P_k \leftarrow (P_i \cdot N_i + P_j \cdot N_j)/(N_i + N_j)$
        $N_k \leftarrow \max(N_i, N_j)$
        Add (P_k, N_k) to C
for each point (P_i, N_i) in C **do**
  Initialize empty set N
  for each point (P_j, N_j) in C **do**
    if distance$(P_i, P_j) < D_{\min}$ **then**
      Add (P_j, N_j) to N
  Remove all points in N from C
  $S \leftarrow$ Mean N_k over (P_k, N_k) in N
  $P \leftarrow$ Weighted mean of P_k in N
  Add (P, S) to C

1.2 Simulation Dataset Construction

To evaluate MHSTGN under scenarios involving station expansion and new deployments, we construct two simulation datasets based on the real-world Hefei dataset by reallocating charging events at the individual event level.

- Hf-Dep

To simulate the deployment of new charging stations, we randomly mask 5% of the original stations and reallocate their traffic to other stations according to a redistribution strategy. Specifically, each EV chooses a neighbor with a 40% probability from geographical neighbors, 30% from preference neighbors, 10% from functional neighbors, 10% from historical neighbors, and 10% from other nodes. The detailed procedure is shown in Algorithm 2.

- Hf-Exp

To simulate expanded stations, we mark 5% of the stations as expanded and set their maximum capacity to half of the original value to represent the pre-expansion capacity. Following the same priority strategy described in Algorithm 2, we reallocate any traffic exceeding the maximum capacity.

Algorithm 2 Charging Event Reallocation Strategy for Hf-Dep (Updated)

Require: Charging events X, masked stations M, unmasked stations $\mathcal{U}$, vehicle histories $\mathcal{H}$, neighbor maps: geographical $\mathcal{N}_g$, preference $\mathcal{N}_p$, functional $\mathcal{N}_f$, historical $\mathcal{N}_h$
Ensure: Reallocated event data X'

Initialize $X' \leftarrow X$
$I \leftarrow$ indices of events in X where station $\in M$
for each index i in I **do**
  $v \leftarrow$ vehicle ID of $X[i]$
  $s \leftarrow$ original station ID of $X[i]$
  Draw random number $r \sim \mathcal{U}(0, 1)$
  if $r < 0.40$ and $\mathcal{N}_g(s) \cap \mathcal{U} \neq \emptyset$ **then**
    $C \leftarrow \mathcal{N}_g(s) \cap \mathcal{U}$
  else if $r < 0.70$ and $\mathcal{N}_p(s) \cap \mathcal{U} \neq \emptyset$ **then**
    $C \leftarrow \mathcal{N}_p(s) \cap \mathcal{U}$
  else if $r < 0.80$ and $\mathcal{N}_f(s) \cap \mathcal{U} \neq \emptyset$ **then**
    $C \leftarrow \mathcal{N}_f(s) \cap \mathcal{U}$
  else if $r < 0.90$ and $\mathcal{N}_h(s) \cap \mathcal{U} \neq \emptyset$ **then**
    $C \leftarrow \mathcal{N}_h(s) \cap \mathcal{U}$
  else
    $C \leftarrow \mathcal{U}$
  $s' \leftarrow$ randomly select a station from C
  Update $X'[i]$ by replacing station s with s'

1.3 Baseline Details

We compare the proposed MHSTGN with a range of baselines, including general spatio-temporal forecasting models and methods specifically designed for EV charging demand prediction:

- **General spatio-temporal forecasting models:**
  - **STGCN** [2]: A spatio-temporal graph convolutional network that models temporal dynamics with CNNs and spatial correlations with GCNs.
  - **ST-SSL** [3]: A self-supervised framework that improves feature learning through two-level adaptive graph augmentations.
  - **DDGCRN** [4]: A recurrent architecture that incorporates dynamic graphs and spatio-temporal embeddings for enhanced forecasting accuracy.

Table 1 Example data samples. The vehicle IDs have undergone desensitization.

	Id	Timestamp	Vehicle state	Charge state	Speed (km/h)	SOC (%)	current (A)	Longitude (°E)	Latitude (°N)
Time-series data	XXXX	2023-11-05 22:30:39	1	3	146	53	10.8	117.394259	31.85702
	XXXX	2023-11-05 22:30:49	1	2	159	53	-4.1	117.394279	31.856534
	...	...	...	...	...	...	...	...	...
	XXXX	2023-11-05 22:32:23	2	1	0	53	-13.0	117.394162	31.856142
	XXXX	2023-11-05 22:32:33	2	1	0	53	-13.2	117.394162	31.856142
	...	...	...	...	...	...	...	...	...
	XXXX	2023-11-06 02:57:53	2	1	0	99	-12.4	117.394162	31.856142
	XXXX	2023-11-06 02:58:03	2	4	0	100	0	117.394162	31.856142
	Id	Charging start time	Charging duration (s)	Starting SOC (%)	Ending SOC (%)	Maximum current (A)	Average current (A)	Longitude (°E)	Latitude (°N)
Charging event	XXXX	2023-11-05 13:22:40	13840	26	67	-13.5	-13.1	117.394142	31.856157
	XXXX	2023-11-05 22:32:23	15930	53	99	-13.4	-12.4	117.394162	31.856142
	...	...	...	...	...	...	...	...	...
	YYYY	2023-11-27 00:30:13	6840	4	100	-172.6	-72.7	117.228762	31.803757
	YYYY	2023-11-28 13:45:00	1530	63	85	-92.4	-80.2	117.207781	31.826258

– **STwave** [5]: A dual-channel network that decomposes traffic signals into trend and event components to better model spatial variations.
– **HGT** [6]: A heterogeneous graph transformer designed to capture complex relational structures through type-specific attention.
– **PDG2Seq** [7]: A Seq2Seq model that integrates periodic feature selection with dynamic graph learning for robust predictions.

• **EV charging demand prediction models:**
– **PAG** [1]: A charging-demand model that builds a physics-inspired graph attention mechanism to capture spatial and temporal patterns.
– **PIAST** [8]: A model that jointly considers charging demand and electricity price signals to characterize user charging behavior.

1.4 Implementation Details

We set the number of layers in the IGEncoder to 2. Both the input and output sequence lengths are fixed at 12. During preprocessing, the data are standardized using Z-score normalization. The number of activated experts k is set to half of the total expert count K. We train our model using the Adam optimizer with an initial learning rate of 0.0001 on an NVIDIA A100 GPU. All datasets are split in a 6:2:2 ratio for training, validation, and testing. And all baseline models are carefully tuned according to the recommended settings to ensure fairness.

1.4.1 Baseline experimental settings

To ensure a fair comparison, all baseline methods are tuned according to their recommended settings. In STGCN [2], the channel dimensions of the three layers in the ST-Conv block are set to 64, 16, and 64. The temporal and spatial convolution kernel sizes in STGCN, ASTGCN [9], and ST-SSL [3] are both set to 3. The DCGRU in DDGCRN [4] uses 64 hidden units, with the number of blocks K set to 2. In STwave [5], the causal convolution kernel size K is set to 2, the DWT level J to 1, the ESGAT scaling factor e to 1, and the spatio-temporal encoder consists of 2 layers. PDG2Seq [7] sets the period embedding dimension $p = 16$, node embedding matrix dimension $d = 8$, and graph convolution depth $K = 2$. Finally, for PAG [1] and PIAST [8], the number of attention heads K and GAT layers M are set to 4 and 2, respectively.

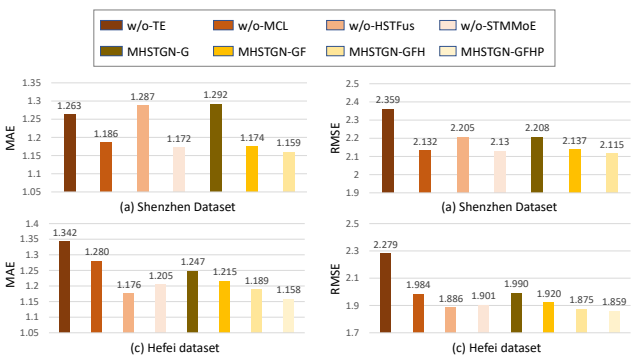


Fig. 1 Performance for ablation models.

1.5 Ablation Study

To assess the impact of individual modules, we evaluate several MHSTGN variants by removing the TE, MCL, and MSTMoE modules. In the w/o-HSTFus variant, hierarchical attention fusion is replaced with a linear fusion approach. We also explore the effect of different subgraphs in MHSTGN: MHSTGN-G uses only the geographic adjacency subgraph, MHSTGN-GF adds functional adjacency, MHSTGN-GFH includes historical adjacency, and MHSTGN-GFHP incorporates all four. The results in Fig. 1 show that removing any key component leads to a performance drop, highlighting the importance of each module. Among them, the time encoding module, which captures periodic temporal patterns, has the most significant impact. And adding multi-view subgraphs further improves accuracy. However, in the Shenzhen dataset, the lack of explicit preference information limits the effectiveness of the preference view.

1.6 Parameter Sensitivity

To analyze the impact of key hyperparameters on the proposed MHSTGN model, we focus on three hyperparameters: λ, K, and M. While adjusting one parameter, the others are kept at their optimal values to ensure consistency. The evaluation results on the Hefei and Shenzhen

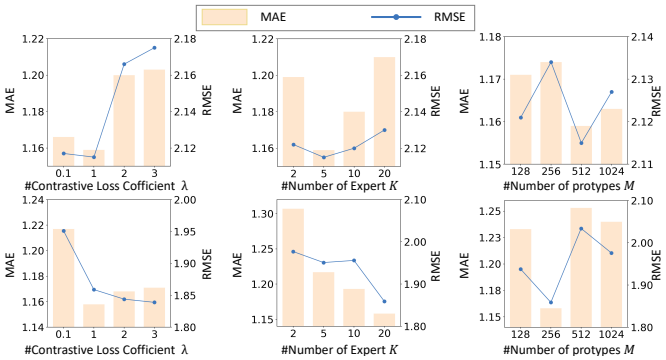


Fig. 2 Parameter sensitivity analysis of MHSTGN. Rows represent Shenzhen (top) and Hefei (bottom) datasets.

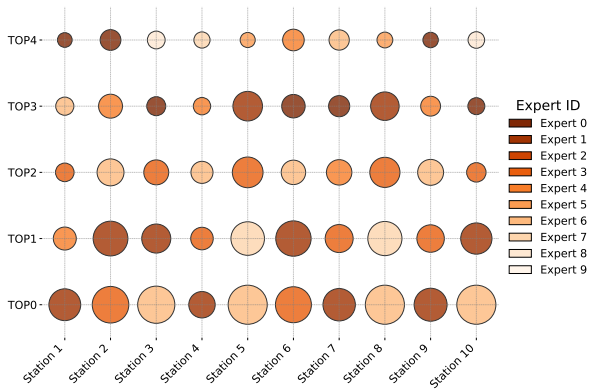


Fig. 3 The top-5 most frequently selected experts by the router in the MMem module for different nodes. The size of the circles represents the selection frequency of each expert.

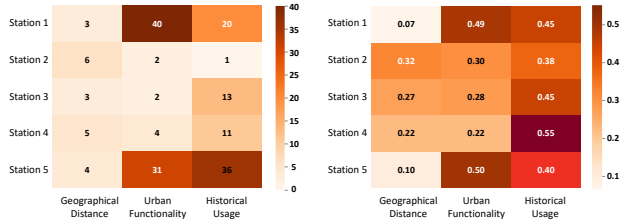
datasets are shown in Fig. 2. Here, λ is the coefficient for contrastive learning, which controls the weight of the contrastive loss relative to the prediction loss. It can be observed that appropriate values of λ significantly enhance model performance. K is the number of experts in the STMMoE module. On the Shenzhen dataset, the optimal K value is 5, while the optimal K value for the Hefei dataset is 20. This might be due to the larger number of stations, longer time span, and more diverse data in the Hefei dataset, requiring more experts to capture different spatio-temporal patterns. M is the number of spatio-temporal prototypes stored in each spatio-temporal memory pool. The combination of K and M reflects the total number of long-term spatio-temporal prototypes required by the STMMoE module.

1.6.1 Effectiveness of Mixture of Experts

To validate the effectiveness of the MOE, we analyze how often each expert is selected by the router for each station on the Hefei test set. In Fig. 3, we randomly select 10 stations for illustration. It is observed that the router dynamically selects different experts for each node. Experts 1, 3, and 6 are chosen most frequently, indicating they effectively capture shared spatio-temporal patterns across different stations.

1.6.2 Effectiveness of Multi-View Subgraphs

To explore the relationship between neighboring nodes and attention scores, we present a case with five stations in the Shenzhen dataset,



(a) # of Neighboring Nodes **(b)** Attention Weights

Fig. 4 The relationship between subgraph neighbor counts and subgraph attention weights.

each having different numbers of neighbors across three subgraphs (Fig. 4(a)). We calculate average attention weights for the subgraphs (Fig. 4(b)). The results indicate that MHSTGN can adaptively select the most relevant subgraph. Stations 1 and 5, with more neighbors in the functional and historical subgraphs, receive higher attention weights, while Stations 3 and 4 emphasize the historical subgraph. Station 2 has balanced weights across all subgraphs due to the limited number of neighbors under each view.

1.7 Case Study

With the rapid growth of charging demand, effective planning of charging station network expansion is crucial for improving service quality and resource utilization. Using the MHSTGN model, we can predict the impact of different expansion schemes on network performance and identify the optimal strategy before deployment. To illustrate the case, we compare two expansion approaches involving about 10% of the stations: one randomly selects nodes for expansion, while the other targets those with consistently saturated utilization.

We evaluate both strategies from local and global perspectives. Locally, we focus on two indicators: the proportion of high-load stations (utilization $\geq$ 90%) and low-load stations (utilization $\leq$ 20%). Under the random expansion strategy, the high-load proportion decreased from 25.33% to 16.59%, while the low-load proportion increased from 14.42% to 22.81%. The targeted expansion further reduced high-load stations from 36.50% to 17.18%, effectively mitigating congestion, but also raised low-load stations from 19.36% to 23.00%. This increase reflects the tidal nature of charging demand, where expansions may lead to greater off-peak underuse. Globally, we measure load balance using the Gini coefficient of average utilization, where lower values indicate better balance. The random expansion achieved a Gini coefficient of 0.1539, while the targeted approach further improved it to 0.1511. Targeted expansion effectively reduces congestion at overloaded stations, but the increase in underused sites suggests the need to pair capacity upgrades with demand-management measures to avoid off-peak inefficiency. These results show that MHSTGN can guide planners in selecting high-impact expansion locations, balancing peak reliability and asset utilization, and making more informed and efficient infrastructure decisions.

References

[1] Qu H, Kuang H, Wang Q, Li J, You L. A physics-informed and attention-based graph learning approach for regional electric vehicle charging demand prediction. IEEE Transactions on Intelligent Transportation Systems, 2024

[2] Yu B, Yin H, Zhu Z. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence. 2018, 3634–3640

[3] Ji J, Wang J, Huang C, Wu J, Xu B, Wu Z, Zhang J, Zheng Y. Spatio-temporal self-supervised learning for traffic flow prediction. In: Proceedings of the AAAI conference on artificial intelligence. 2023, 4356–4364

[4] Weng W, Fan J, Wu H, Hu Y, Tian H, Zhu F, Wu J. A decomposition dynamic graph convolutional recurrent network for traffic forecasting. Pattern Recognition, 2023, 142: 109670

[5] Fang Y, Qin Y, Luo H, Zhao F, Xu B, Zeng L, Wang C. When spatio-temporal meet wavelets: Disentangled traffic forecasting via efficient spectral graph attention networks. In: 2023 IEEE 39th International Conference on Data Engineering (ICDE). 2023, 517–529

[6] Hu Z, Dong Y, Wang K, Sun Y. Heterogeneous graph transformer. In: Proceedings of the web conference 2020. 2020, 2704–2710

[7] Fan J, Weng W, Chen Q, Wu H, Wu J. Pdg2seq: Periodic dynamic graph to sequence model for traffic flow prediction. Neural Networks, 2024, 106941

[8] Kuang H, Qu H, Deng K, Li J. A physics-informed graph learning approach for citywide electric vehicle charging demand prediction and pricing. Applied Energy, 2024, 363: 123059

[9] Guo S, Lin Y, Feng N, Song C, Wan H. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In: Proceedings of the AAAI conference on artificial intelligence. 2019, 922–929