

Transformer-Driven Dynamic Forecasting and Scheduling Optimization of Tourist Flow

Xiaona Xie¹, Wenliang Zhang²

¹Department of Teaching Management, Henan College of Transportation, Zhengzhou 450000, Henan, China

²Department of Science & Technology, Zhengzhou University, Zhengzhou 450001, Henan, China

Corresponding author: Wenliang Zhang (e-mail: 18603718969@163.com)

ABSTRACT Tourist flow modeling and prediction in scenic areas is highly complex, exhibiting significant spatiotemporal dependence and being influenced by various external factors. Traditional models struggle to simultaneously characterize complex spatiotemporal relationships and integrate diverse external information, posing challenges for intelligent resource management and dynamic information scheduling in large-scale networked systems. To address this, this paper proposes a closed-loop integration framework that combines a particle swarm optimization (PSO)-based Temporal Fusion Transformer-Graph Attention Network (TFT-GAT) prediction model with Deep Double-Q Network (D3QN) scheduling optimization. The framework integrates heterogeneous data such as historical traffic, weather, and social media, achieving adaptive time-varying spatial embedding through a graph attention network (GAT) while capturing long- and short-term dependencies using a Temporal Fusion Transformer (TFT) for both point and quantile prediction. The PSO algorithm performs global optimization of the TFT-GAT hyperparameters, and the resulting prediction outputs together with uncertainty estimates are incorporated into the D3QN to realize closed-loop online capacity allocation based on reinforcement learning. Such a data-driven spatiotemporal modeling strategy also provides methodological insights for dynamic information fusion and adaptive resource scheduling in intelligent electromagnetic sensing and communication environments. Experiments conducted at the Forbidden City in Beijing demonstrate high prediction accuracy with a mean absolute error of 1.5–2.6 people/hour, an average quantile coverage exceeding 82%, and a response time of 140.6 ms. The incorporation of exogenous factors, particularly holidays, improves prediction performance by 18.4%, validating the proposed framework's robustness in spatiotemporal coupling, uncertainty representation, and real-time scheduling.

INDEX TERMS tourist flow prediction, temporal fusion transformer-graph attention network, deep reinforcement learning, spatiotemporal information fusion, intelligent electromagnetic sensing

I. INTRODUCTION

THE rapid expansion of the tourism industry and the diversification of travel patterns have caused visitor flow to scenic spots to exhibit a high degree of spatiotemporal nonstationarity, characterized by both short-term peaks and pronounced holiday or seasonal fluctuations. Consequently, accurately forecasting and managing capacity in both the tourism sectors requires analytical models capable of robustly handling these volatile, non-stationary temporal and spatial dependencies [1-3]. Furthermore, complex attraction-spillover effects exist between different attractions. However, in reality, visitor flow is influenced by multiple exogenous factors, such as weather, holidays, events, traffic conditions, and online popularity [4,5]. Transferring reinforcement learning methods from transportation or logistics scenarios to scenic spot scheduling presents distinct challenges. These scenarios exhibit

significant differences in state-action space, temporal scales, constraints and safety requirements, sample availability, and the gap between simulation and the real world. This results in low training sample efficiency, strategies prone to overfitting in simulation environments, and insufficient adaptability to safety and regulatory constraints. Therefore, when introducing reinforcement learning for scenic spot scheduling, domain adaptation, sample-efficient algorithms, and verifiable safety projection mechanisms must be considered.

This paper aims to address the inaccuracy and lag in scenic spot visitor flow forecasting and scheduling by constructing an intelligent optimization framework that integrates forecasting and scheduling. In experiments, this framework significantly outperformed existing methods on weekdays, weekends, and holidays. The mean absolute error (MAE) for predictions on statutory holidays was only

2.6 people/hour, a reduction of 13 people/hour compared to the traditional CNN-BiLSTM-Attention (Convolutional Neural Network-Bidirectional Long Short-Term Memory-Attention) model. Percentile coverage was approximately 86.3% on weekdays, 85.1% on weekends, and 82.7% on holidays. In scheduling, average waiting time was reduced to 4.2 minutes, more than half that of rule-based scheduling, and response time was approximately 140.6 milliseconds. Extending this finding, the demonstrated capability for robust prediction and optimized resource allocation carries direct implications for industrial supply chains. Specifically, by mitigating uncertainty and optimizing the flow of resources, this approach provides a scalable, intelligent blueprint for improving inventory control, enhancing responsiveness to volatile fashion demands, and ultimately boosting resource utilization efficiency across the complex supply chain.

II. RELATED WORK

In the field of dynamic prediction of tourist flow, scholars have proposed a variety of methods from various perspectives. Yu J used differential evolution and gray wolf optimization to optimize the parameters of the LSTM (Long Short-Term Memory) scenic spot short-term tourist flow prediction model, significantly improving the prediction accuracy ($R^2=0.945$). However, it still mainly relies on traditional time series modeling and lacks spatial dependence and the fusion of exogenous multi-source information [6]. Tang Q's MACBL (multi-attentional convolutional bidirectional long short-term memory) can reduce RMSE (Root Mean Square Error)/MAE/MAPE (Mean Absolute Percentage Error) at multiple time scales, but it mainly strengthens temporal and local features and does not systematically model network-level spatial interactions [7]. Overall, existing methods have made breakthroughs in the introduction of spatial structure, grid modeling, and lag interaction mining, but they generally rely on static or preset metrics and are still insufficient in adaptive modeling of dynamic and multiscale spatial dependencies.

III. INTEGRATED FRAMEWORK FOR DYNAMIC PREDICTION AND SCHEDULING OF TOURIST FLOW MULTI-SOURCE DATA FUSION

To accurately predict and schedule tourist flow, this paper fuses multi-source heterogeneous data from scenic spots. The fused data types mainly include historical passenger flow data, meteorological environmental factors, holiday and event factors, transportation accessibility factors, and social media popularity factors. The actual inbound passenger flow at the scenic spot at node $\hat{v}$ and time t is defined as $x_{\hat{v},t}^{kl}$. Due to the non-stationary nature of passenger flow, sliding window normalization is used to obtain $\tilde{x}_{\hat{v},t}^{kl}$. The definition of meteorological environmental factors is shown in formula (1).

$$x_t^{(\omega)} = \left[x_t^{(\text{temp})}, x_t^{(\text{humid})}, x_t^{(\text{rain})}, x_t^{(\text{wind})} \right]^T \quad (1)$$

In formula (1), $x_t^{(\text{temp})}$ represents temperature, $x_t^{(\text{humid})}$ represents humidity, $x_t^{(\text{rain})}$ represents rainfall, and $x_t^{(\text{wind})}$ represents wind speed. Considering the nonlinear impact of weather on passenger flow, dimension-by-dimension Z-score normalization is used to obtain $\tilde{x}_t^{(\omega)}$.

The holiday and event factors are defined as $\tilde{x}_t^{(\text{ho})} = [x_t^{(\text{ho})}, x_t^{(\text{eo})}]$, where $x_t^{(\text{ho})}$ is 1 if it falls on a statutory holiday or major event day, and 0 otherwise. The activity level $x_t^{(\text{eo})}$ is one-hot encoded.

In the definition of the traffic accessibility factor, the traffic accessibility vector is defined as $x_t^{(\text{tr})}$. Furthermore, the road network travel time $\tau_{ro,t}$ is introduced, and the traffic congestion index is defined as $c_{ro,t} = \tau_{ro,t} / \tau_{ro}^{\min} \geq 1$, where $\tau_{ro}^{\min}$ is the minimum travel time under unobstructed conditions. The final traffic characteristic is expressed as shown in formula (2).

$$\tilde{x}_t^{(\text{tr})} = \left[\frac{x_{ro,t}^{(\text{tr})}}{\max_t x_{ro,t}^{(\text{tr})}}, \frac{1}{c_{ro,t}} \right]^{|\text{RO}|} \quad (2)$$

In formula (2), RO represents different access channels to the park (subway, bus, self-driving highway, scenic area special line, etc.).

This paper crawls the Weibo/WeChat/TikTok popularity index of scenic area-related keywords, and the definition formula is shown in formula (3).

$$x_t^{(\text{sw})} = \log(1 + \text{count}(\text{keyword}, t)) \quad (3)$$

Formula (3) represents the number of relevant posts. Considering the time lag effect, smooth convolution is introduced for processing, resulting in $\hat{x}_t^{(\text{sw})}$. At time step t , the multi-source features are concatenated to obtain the formula shown in (4).

$$X_t = [\tilde{x}_{\hat{v},t}^{kl}, \tilde{x}_t^{(\omega)}, \tilde{x}_t^{(\text{ho})}, \tilde{x}_t^{(\text{tr})}, \tilde{x}_t^{(\text{sw})}]^T \quad (4)$$

To capture the importance of different factors, attention weights are introduced for weighting, and the input vector after weighted fusion is U_t .

GAT SPATIAL RELATIONSHIP EMBEDDING

In order to capture the spatial interaction relationship between different scenic spots, attractions, and entrances and exits, this paper constructs a spatial relationship embedding module based on GAT [8,9]. In this study, the nodes of GAT are the sub-regions of the scenic area (such as entrances, core attractions, and functional areas). The initial edges between nodes are determined by two types of information. Based on the spatial accessibility of the road network (i.e., proximity relationships reflected by the shortest geographical path and travel time) and historical visitor flow statistics (the frequency and correlation of visitor flow exchange between nodes), these two parts of information are fused according to weights to construct an adjacency

matrix and dynamically updated by time slices. Subsequently, GAT's learnable attention mechanism adaptively amplifies or suppresses the influence of each edge during the training process, so that the model can reflect both the proximity of physical space and capture the actual tourist flow and spillover effect, thus characterizing time-varying spatial dependence.

GRAPH STRUCTURE CONSTRUCTION AND NODE FEATURE INPUT

First, each sub-area in the scenic area (such as entrance, core attraction, functional area) is regarded as a node set $\hat{V}$ in the graph. The spatial connection relationship between any two nodes is represented by the adjacency matrix B , and B is defined as shown in formula (5).

$$B_{ij} = \begin{cases} \exp\left(-\frac{\text{dis}_{ij}^2}{\beta^2}\right), & i \neq j \\ 0, & i = j \end{cases} \quad (5)$$

In formula (5), dis_{ij}^2 represents the geographic shortest path distance between two nodes, and β represents the Gaussian kernel scale parameter.

The input features of each node are shown in formula (6).

$$h_i^{(0)}(t) = W_0 u_i^{(t)} + b_0 \quad (6)$$

In formula (6), W_0 , b_0 represent linear transformation parameters.

ATTENTION WEIGHT CALCULATION

GAT adaptively models spatial dependencies through a learnable attention mechanism. The attention weight calculation is shown in formula (7).

$$e_{ij}(t) = \text{LeakyReLU}\left(a^\top \left[Wh_i^{(0)}(t) \| Wh_j^{(0)}(t)\right]\right) \quad (7)$$

In formula (7), W represents the feature linear transformation matrix, and $a^\top$ represents the attention parameter vector.

The normalized neighbor attention weight is shown in formula (8).

$$\alpha_{ij}(t) = \frac{\exp(e_{ij}(t))}{\sum_{k \in N(i)} \exp(e_{ik}(t))} \quad (8)$$

In formula (8), $\alpha_{ij}(t)$ represents the normalized neighbor attention weight.

SPATIAL EMBEDDING UPDATE

The spatial embedding representation of a node is updated as shown in formula (9).

$$\mathbf{h}_i^{(1)}(t) = \left\| \sum_m \left(\sum_{j \in \square(i)} \alpha_{ij}^{(m)}(t) \mathbf{W}^{(m)} \mathbf{h}_j^{(0)}(t) \right) \right\| \quad (9)$$

In formula (9), M represents the number of attention heads, and $\mathbf{W}^{(m)}$ and $\alpha_{ij}^{(m)}(t)$ represent the parameters and weights of the attention heads, respectively. Finally, the computation results for all nodes at all time slices are stacked to obtain the GAT spatial embedding sequence $H^{(1)}$.

The spatial relationship diagram is shown in Figure 1.

Figure 1 shows a spatial relationship diagram of the Forbidden City. The red squares represent the four gates, and the blue dots represent core scenic spots. The edge color and line type reflect spatial weights: thick solid red lines represent high-weight edges, solid orange lines represent medium-weight edges, and dashed gray lines represent low-weight edges. The links to major scenic spots along the main axis have slightly higher weights with slight variations, reflecting the centrality of these attractions. The connection between the Meridian Gate and the main axis has a lower weight, indicating the relative directionality of the entrance to the core scenic area. The connection from Shenwu Gate/Donghua Gate to the Imperial Garden is assigned a higher weight, highlighting the primary route likely used by tourists.

TFT TIME SERIES PREDICTION AND OUTPUT LAYER DESIGN

INPUT ENCODING AND VARIABLE SELECTION

TFT is based on the Transformer model [10-12]. After completing multi-source data fusion and spatial relationship embedding, this paper obtains the fused input sequence $\hat{X} = U_t \oplus H_t^{(1)}$. First, linear projection and variable selection are performed on the input sequence to screen effective features, as shown in formula (10).

$$\begin{cases} \dot{X}_t = W_{\dot{v}s} \hat{X}_t + b_{\dot{v}s}, \\ \alpha_{1t} = \text{Softmax}(W_{\alpha 1} \dot{X}_t), \\ Z_t = \alpha_{1t} \odot \dot{X}_t. \end{cases} \quad (10)$$

In formula (10), α_{1t} represents the feature selection weight, Z_t represents the final input representation, and $\odot$ represents element-by-element multiplication.

LSTM LAYER EXTRACTS TEMPORAL DEPENDENCIES

In TFT, local temporal dependencies are first extracted through gated recurrent units [13]. This paper uses a single-layer LSTM to model Z_t , as shown in formula (11).

$$\hat{h}_t = \text{LSTM}(Z_t, \hat{h}_{t-1}) \quad (11)$$

In formula (11), $\hat{h}_t$ represents the hidden state at time t . The LSTM state update formula is shown in (12).

$$\begin{aligned} f_t &= \sigma(W_f Z_t + \hat{U}_f h_{t-1} + b_f), \\ \gamma_t &= \sigma(W_\gamma Z_t + \hat{U}_\gamma h_{t-1} + b_\gamma), \\ o_t &= \sigma(W_o Z_t + \hat{U}_o h_{t-1} + b_o), \\ \tilde{d}_t &= \tanh(W_d Z_t + \hat{U}_d h_{t-1} + b_d), \\ d_t &= f_t \odot d_{t-1} + \gamma_t \odot \tilde{d}_t, \\ \hat{h}_t &= o_t \odot \tanh(d_t). \end{aligned} \quad (12)$$

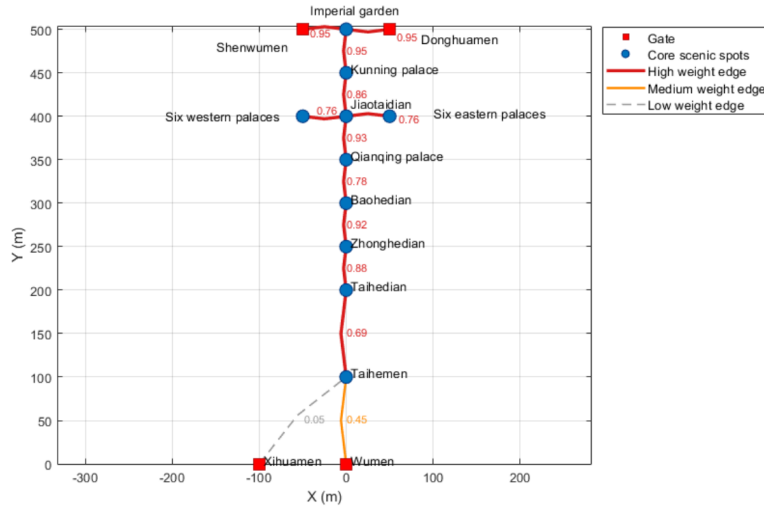


FIGURE 1. Spatial Relationship Diagram

In formula (12), σ represents the Sigmoid function, and d_t represents the cell state.

MULTI-HEAD ATTENTION MECHANISM EXTRACTS GLOBAL DEPENDENCIES

After capturing local dependencies, the LSTM output is input into the multi-head attention layer to obtain long-term temporal dependencies. For each attention head, the formula is shown in (13).

$$Q^{(m)} = \hat{h}W_Q^{(m)}, \quad K^{(m)} = \hat{h}W_K^{(m)}, \quad V^{(m)} = \hat{h}W_V^{(m)} \quad (13)$$

In formula (13), $Q^{(m)}$, $K^{(m)}$, $V^{(m)}$ represent the query, key, and value matrices, respectively.

The attention weight is calculated as shown in formula (14).

$$\text{Attn}^{(m)}(Q^{(m)}, K^{(m)}, V^{(m)}) = \text{Softmax} \left(\frac{Q^{(m)}K^{(m)\top}}{\sqrt{d_q}} \right) \times V^{(m)} \quad (14)$$

In formula (14), d_q represents the dimension of each head. $\hat{H}_{att}$ is obtained by concatenating the multi-head results.

GATED RESIDUAL NETWORK AND SKIP CONNECTION

To suppress invalid features and avoid overfitting, a gated residual network (GRN) is used to perform nonlinear transformation on the concatenated multi-head results, as shown in formula (15).

$$\hat{H} = \delta(\hat{H}_{att}) \odot \tanh(W_g \hat{H}_{att} + b_g) + (1 - \delta(\hat{H}_{att})) \odot \hat{H}_{att} \quad (15)$$

In formula (15), δ represents the gating function, and $\hat{H}$ represents the result after nonlinear transformation.

OUTPUT LAYER DESIGN

For the prediction layer, the predicted value is shown in formula (16).

$$\hat{y}_{t+\pi}^{(0)} = \hat{f}_{\theta}(\hat{H}_t) \quad (16)$$

In formula (16), $\hat{f}_{\theta}$ represents the fully connected layer mapping, θ represents the model parameters, and π represents the future prediction step size.

To support quantile prediction, a multi-head mapping is used in the output layer, corresponding to each quantile ι_k . For each quantile, the predicted value is shown in formula (17).

$$\hat{y}_{t,\pi}^{(\iota_k)} = W_{\iota_k} \hat{H}_t + b_{\iota_k} \quad (17)$$

In formula (17), the ι_k quantiles are 0.1, 0.5, and 0.9, and W_{ι_k} and b_{ι_k} represent the parameters corresponding to these quantiles. Within the framework of this paper, quantile forecasting is used to characterize the uncertainty of tourism traffic, providing information support for risk perception in scheduling strategies. Different quantiles correspond to potential passenger flow fluctuation ranges, with the median reflecting the most likely passenger flow level, while high and low quantiles represent extreme peak and trough scenarios, respectively. Based on this, the D3QN scheduling module can formulate differentiated strategies at different confidence levels, such as increasing capacity or opening backup channels in advance under high quantile forecasts, effectively avoiding congestion risks and improving the robustness of emergency response.

The integrated framework for dynamic prediction and scheduling of tourist flow is shown in Figure 2.

Figure 2 illustrates the integrated framework for dynamic forecasting and scheduling optimization of tourist flow. The overall process consists of three stages: data processing, predictive modeling, and scheduling optimization. Multi-source

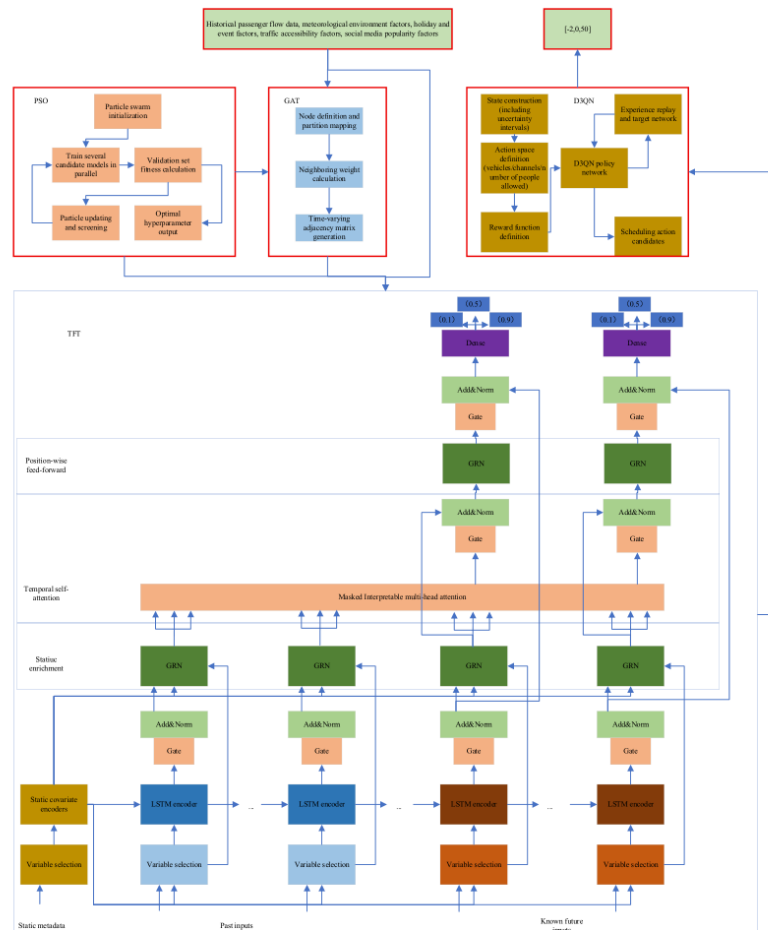


FIGURE 2. Integrated Framework for Dynamic Forecasting and Scheduling of Tourist Flow

data fusion integrates external information such as scenic area visitor flow, traffic, weather, and holidays. Spatial relationship embedding based on GAT captures the complex interactions between different attractions and regions. TFT is then used to perform multi-quantile forecasting, combined with the PSO algorithm to optimize model hyperparameters. The prediction results are then fed into the D3QN scheduling optimization module, which dynamically decides on tourist diversion and resource allocation. A hybrid training mechanism is then used to achieve collaborative optimization of prediction and scheduling, forming a closed-loop, integrated intelligent framework.

PSO HYPERPARAMETER OPTIMIZATION

To improve the stability and generalization capabilities of the TFT-GAT model in tourist flow prediction, PSO is used to perform a global search and adaptive update of key hyperparameters [14,15]. The parameters to be optimized include the number of GAT layers, the hidden dimension of the TFT attention layer, the negative slope of the LeakyReLU, the dropout probability, the learning rate, the batch size, the number of TFT encoder layers, and the dimension of the fully connected layer.

In this paper, Particle Swarm Optimization (PSO) is used

to globally tune the key hyperparameters of the TFT-GAT model to improve prediction performance and generalization ability. Specifically, the optimized hyperparameters include the number of GAT layers, the hidden layer dimension of the TFT attention layer, the negative slope of LeakyReLU, the dropout probability, the learning rate, the batch size, the number of TFT encoder layers, and the dimension of the fully connected layers. The global tuning range defines the value range for each hyperparameter, such as 1–4 GAT layers, 64–256 hidden dimension of the attention layer, 0.1–0.5 slope of LeakyReLU, 0.1–0.5 dropout probability, $1e-5$ – $1e-3$ learning rate, 32–128 batch size, 2–4 TFT encoder layers, and 128–512 dimension of the fully connected layers. PSO iteratively updates particle positions and velocities within this global search space, evaluates fitness using prediction errors on the validation set, and ultimately finds the optimal combination of hyperparameters to achieve the globally optimal configuration of the TFT-GAT model.

The position vector of the j th particle is defined as $\mathbf{s}_j$, where each subset represents the value of the j th particle in the ζ th hyperparameter dimension, and the velocity vector is defined as $\mathbf{d}_j$. The initialization operation is shown in formula (18).

$$\begin{cases} dx_{j\zeta}^{(0)} \sim U(\theta_{\zeta}^{\min}, \theta_{\zeta}^{\max}), \\ dv_{j\zeta}^{(0)} \sim U(-\Delta_{\zeta}, \Delta_{\zeta}). \end{cases} \quad (18)$$

In formula (18), U represents a uniform distribution, and $\theta_{\zeta}^{\min}$ and $\theta_{\zeta}^{\max}$ represent the minimum and maximum values of the hyperparameters, respectively.

The particle fitness is measured by the prediction loss on the validation set, which is defined as T_{val} . Given the hyperparameter configuration dxj , TFT-GAT is trained and the prediction output $YD_{val} = \{(\text{Ut}, y_t)\}_{t=1}$ is calculated. The loss function is the weighted combination error, as shown in formula (19).

$$\begin{aligned} \check{f}(dx_j) = & \lambda_1 \cdot \frac{1}{T_{val}N} \sum_{t=1}^{T_{val}} \|y_t - \hat{y}_t^{(j)}\|_2^2 \\ & + \lambda_2 \cdot \frac{1}{T_{val}N} \sum_{t=1}^{T_{val}} |y_t - \hat{y}_t^{(j)}| \end{aligned} \quad (19)$$

In formula (19), λ_1 and λ_2 represent weight coefficients, and y_t represents the actual passenger flow. At iteration step t , the speed and position update rules are as shown in formula (20) [16,17], and boundary constraints are applied after the update.

$$\begin{aligned} dv_{jk}^{(t+1)} = & \chi [dv_{jk}^{(t)} + \hat{c}_1 \hat{r}_1 (p_k^{best} - dx_{jk}^{(t)}) \\ & + \hat{c}_2 \hat{r}_2 (g_k^{best} - dx_{jk}^{(t)})], \\ dx_{jk}^{(t+1)} = & dx_{jk}^{(t)} + dv_{jk}^{(t+1)}. \end{aligned} \quad (20)$$

In formula (20), p_k^{best} represents the historical optimal position, g_k^{best} represents the global optimal position, $\hat{r}_1$ and $\hat{r}_2$ represent random weights, $\hat{c}_1$ and $\hat{c}_2$ represent individual and group learning factors, and χ represents the shrinkage factor. Early termination occurs if one of the following conditions is met: $\min_j \check{f}(x_j^{(t)}) \leq \epsilon$ or $t \geq T_{max}$, where ϵ is the preset threshold. The final hyperparameter solution is $\theta^* = \arg \min_{x_j \in \Omega} \check{f}(dx_j)$.

D3QN SCHEDULING OPTIMIZATION

STATE SPACE CONSTRUCTION

This paper chooses D3QN instead of the simpler RL algorithm, mainly because of the noise and uncertainty in scheduling scenarios. The prediction model output brings time-varying and biased environmental information, and using a single estimator can easily lead to excessively high Q-value estimates and policy instability. D3QN's dual estimator can effectively suppress this overestimation bias, improve the accuracy of Q-value estimation and the stability of the training process. At the same time, its value-based off-policy learning (combining experience replay and target network) is more efficient for offline historical data and online sparse interactions, and is naturally adapted to our defined discrete adjustment action space (vehicles/tickets/number of passengers allowed to pass).

In the closed-loop integration framework of this paper, the prediction results of the TFT-GAT model (including future passenger flow at different quantiles) are directly used as part of the input to the state space of the D3QN scheduling module. D3QN uses this prediction information to assess the potential congestion and resource demand of the current scenic area, and dynamically selects the optimal diversion measures and resource scheduling strategies in the discrete action space, thereby achieving tight coupling and collaborative optimization between prediction and scheduling.

This study adopts the D3QN scheduling optimization model to achieve adaptive allocation and dynamic scheduling of tourist flow and transportation resources [18]. The state space is defined as shown in formula (21).

$$s_t = [\hat{y}_t, \hat{y}_t^{(0.1)}, \hat{y}_t^{(0.5)}, \hat{y}_t^{(0.9)}, CL_t, QL_t, EL_t] \quad (21)$$

In formula (21), $\hat{y}_t^{(0.1)}$, $\hat{y}_t^{(0.5)}$, and $\hat{y}_t^{(0.9)}$ represent the 10%, 50%, and 90% percentile predicted passenger flows, respectively; CL_t represents the currently dispatchable capacity (e.g., number of vehicles and security checkpoints); QL_t represents the cumulative congestion index (maximum carrying capacity) within the scenic area; and EL_t represents exogenous environmental disturbance factors (numerical results such as weather levels and holiday markings).

ACTION SPACE DEFINITION

The state space is defined as shown in formula (22).

$$\hat{A}_t = \{\Delta car_t, \Delta td_t, \Delta Person_t\} \quad (22)$$

In formula (22), Δcar_t represents the increase or decrease in the number of buses or subway vehicles at time t , with Δtd_t values of -2, -1, 0, 1, and 2. Δtd_t represents the adjustment of security check/ticket inspection channels, with values of -1, 0, and 1. $\Delta Person_t$ represents the dynamic adjustment of the number of people released for time-sharing ticketing, with values of -50, 0, and 50, in units of people per 5 minutes.

REWARD FUNCTION DESIGN

To guide the model to balance "congestion control" and "capacity cost" during peak passenger flow, a reward function is constructed as shown in formula (23).

$$\begin{aligned} r_t = & -\lambda_3 \cdot \max \left(0, \frac{QL_t}{QL_{max}} - \theta \right)^2 - \lambda_4 \cdot \|\Delta car_t\| \\ & - \lambda_5 \cdot \|\Delta td_t\| - \lambda_6 \cdot \|\Delta Person_t\| \end{aligned} \quad (23)$$

In formula (23), θ represents the allowed safety threshold (0.7), and λ_3 , λ_4 , λ_5 , λ_6 represent the corresponding penalty weights.

D3QN NETWORK STRUCTURE AND UPDATE

D3QN decomposes the Q value into the state value function $\hat{V}$ and the action advantage function $\hat{A}$. The structure is shown in formula (24).

$$Q(s_t, \hat{a}; \theta) = \hat{V}(s_t; \theta_{\hat{V}}) + \left(A(s_t, \hat{a}; \theta_{\hat{A}}) - \frac{1}{|\hat{A}|} \sum_{\hat{a}'} \hat{A}(s_t, \hat{a}'; \theta_{\hat{A}}) \right) \quad (24)$$

In formula (24), $\theta_{\hat{V}}$ represents the value branch parameter, and $\theta_{\hat{A}}$ represents the advantage branch parameter.

To reduce the overestimation bias, the Double Q-Learning update rule is introduced, as shown in formula (25).

$$y_t = r_t + \varpi Q \left(s_{t+1}, \arg \max_{\hat{a}'} Q(s_{t+1}, \hat{a}'; \theta); \theta^- \right) \quad (25)$$

In formula (25), ϖ represents the discount factor, θ^- represents the target network parameter, and is updated synchronously with the online network parameter θ every fixed number of steps. The loss function is defined as the mean squared error, as shown in formula (26).

$$Loss_{D3QN}(\theta) = \mathbb{E}_{(s_t, \hat{a}_t, r_t, s_{t+1}) \sim D} \left[\left(y_t - Q(s_t, \hat{a}_t; \theta) \right)^2 \right] \quad (26)$$

In formula (26), D represents the experience pool. The D3QN process is as follows: (1) Experience pool, the output of the prediction model TFT-GAT is used as the environment input, and the D3QN network parameters are randomly initialized. (2) At time t , the action is selected according to the ϵ -greedy strategy to perform the scheduling operation, obtain the reward r_t and the next state s_{t+1} , and store them in the experience pool; (3) Randomly extract batch data from the experience pool and perform parameter updates. (4) Copy θ to θ^- every fixed number of steps. (5) Stop training when the cumulative reward converges or the average congestion is lower than the threshold.

LOSS FUNCTION DESIGN

After the independent construction of the TFT-GAT prediction module and the D3QN scheduling optimization module, the overall training process adopts an end-to-end hybrid optimization strategy. This strategy integrates prediction accuracy and scheduling benefit into a unified objective through a joint loss function and uses the Adam optimizer for parameter updates [19-21].

The loss function for the prediction task uses the quantile regression loss, as shown in formula (27).

$$\begin{cases} Loss_{predict} = \sum_k^K \lambda_7 Loss_{\iota_k}(y, \hat{y}^{(\iota_k)}), \\ Loss_{\iota_k}(y, \hat{y}^{(\iota_k)}) = \max \left[\iota_k(y - \hat{y}^{(\iota_k)}), \right. \\ \quad \left. (\iota_k - 1)(y - \hat{y}^{(\iota_k)}) \right]. \end{cases} \quad (27)$$

In formula (27), $Loss_{predict}$ represents the prediction loss and λ_7 represents the weight factor. The total loss function is shown in formula (28).

$$Loss_{total} = \lambda_8 Loss_{predict} + \lambda_9 Loss_{D3QN} \quad (28)$$

In formula (28), λ_8 and λ_9 represent the loss weight hyperparameters.

The hyperparameter settings are shown in Table 1.

TABLE 1. Hyperparameters

Module	Parameter	Value	Module	Parameter	Value
GAT	Number of layers	2	PSO	Population size	30
GAT	Number of attention heads	8	PSO	Maximum number of iterations	50
GAT	LeakyReLU negative slope	0.2	PSO	Inertia weight	0.7
GAT	Dropout probability	0.3	PSO	Individual and group learning factors	1.5
TFT	Attention hidden dimension	128	D3QN	Discount factor	0.99
TFT	Number of encoder layers	3	D3QN	Experience replay pool size	10000
TFT	Fully connected layer dimension	256	D3QN	Target network update frequency	200
TFT	Dropout probability	0.2	D3QN	Exploration rate	1.0 linearly decays to 0.05
Training optimization	Learning rate	1×10^{-4}	Training optimization	Maximum number of training epochs	200
Training optimization	Batch size	64	Training optimization	Early stopping strategy	Stop if no improvement is seen on the validation set for 20 consecutive rounds

IV. EXPERIMENT ON DYNAMIC PREDICTION OF TOURIST FLOW

EXPERIMENTAL DATA

This study focuses on the Palace Museum in Beijing. The spatial division includes four entrance gates (Wumen, Shenwumen, Donghuamen, and Xihuamen) and 10 core scenic spots (Taihemen, Taihedian, Zhonghedian, Baohe-dian, Qianqing Palace, Jiaotaidian, Kunminggong, Imperial Garden, East Six Palaces, and West Six Palaces). Data were collected from January to December 2024, aggregated and aligned on an hourly basis, resulting in a total of 122,640 sets of data. Historical visitor flow data comes from the Palace Museum's access control system and electronic ticketing system (including gate entry and exit records and online ticketing transactions). Meteorological environmental factors come from the Beijing Meteorological Observatory (hourly observations). Holiday and event factors come from the national statutory holiday schedule, the Palace Museum's event calendar (official announcements), and records of special events at the museum. Accessibility factors come from the Beijing Municipal Transportation Administration. Social media popularity factors are derived from crawled posts/searches on public social media platforms (Weibo, TikTok, and WeChat public platforms) for keywords such as

"Palace Museum," "Forbidden City," and "Midnight Gate." Before integrating social media data into the TFT-GAT model, the crawled posts underwent text cleaning, including removing noise characters, stop words, and duplicate information. Subsequently, the text was quantified using keyword matching, sentiment analysis, and topic modeling (LDA), transforming user attention and sentiment into numerical features such as the daily number of relevant posts, the positive/negative sentiment ratio, and the intensity of trending topics. After time smoothing, these features were used as heterogeneous input and fused with other multi-source data to capture the potential impact of social media on tourist flow in scenic areas.

EVALUATION METRICS

The formula for MAE is shown in (29):

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (29)$$

In formula (29), y_i represents the true value and $\hat{y}_i$ represents the predicted value.

The formula for RMSE is shown in (30):

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (30)$$

The formula of MAPE is shown in (31):

$$MAPE = \frac{100\%}{N} \sum_{i=1}^N \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (31)$$

EXPERIMENTAL PLAN

The experimental plan is as follows: First, the paper maintains strict forward validation of the time series during data segmentation and training. The data is divided into a training set (70%), a validation set (15%), and a test set (15%) in chronological order. All models are trained and evaluated using the same training/validation/test split to ensure comparability. Predictor models compared include: SAE-LSTM, STC-LSTM, CNN-BiLSTM-Attention, ASTGCN (Attention Based Spatial-Temporal Graph Convolutional Networks), GCN-RNN, Informer-LSTM, Autoformer, and PSO-TFT-GAT model. Hyperparameters for each model are tuned using the validation set (the baseline model uses Bayesian search to ensure competitiveness). The hyperparameters of the PSO-TFT-GAT model were obtained through a PSO search. Training details were standardized: the Adam optimizer, batch training, early stopping, and the same random seed set were used (10 different seeds were tested, and the mean and standard deviation were reported). Evaluation metrics covered predictive performance (MAE, RMSE, MAPE, CRPS, PICP) as well as subsequent scheduling metrics (average waiting time, traffic flow, congestion duration, and operating costs). These metrics were analyzed for four scenarios: regular days, peak holiday periods,

and extreme weather and emergencies, to demonstrate the model's robustness.

V. EXPERIMENTAL RESULTS

Overall Prediction Accuracy (Point Estimate)

Figure 3 shows a comparison of the actual and predicted passenger flow curves for different models.

In Figure 3, the overall trend shows that all models effectively capture the two peak periods of passenger flow, around 9:30 AM and 1:30 PM. Predictions fluctuate less during low periods (such as around noon and in the late afternoon), and the curve trends generally align with actual data. On weekdays, due to the strong regularity of passenger flow and the generally high prediction accuracy of the models, the curves closely match the actual data. During weekends and statutory holidays, forecast errors increase significantly due to uncertainties, and models are particularly prone to underestimation or overestimation during peak periods. The overall prediction accuracy is shown in Figure 4.

In Figure 4 (a), the PSO-TFT-GAT model achieves extremely low absolute error on weekdays, with a MAE of 1.5 people/hour, RMSE of 1.3 people/hour, and MAPE of 0.7%. This is significantly better than the Informer-LSTM (MAE of 4.6 people/hour, RMSE of 3.7 people/hour, and MAPE of 1.8%) and the Autoformer (MAE = 3.0 people/hour, RMSE = 2.3 people/hour, and MAPE = 0.9%). The PSO-TFT-GAT model achieved an RMSE 2.4 people/hour lower than the Informer-LSTM model and 1.0 people/hour lower than the Autoformer model. The SAE-LSTM model achieved an RMSE of 12.0 people/hour. This indicates that in weekday scenarios, this method significantly improves both absolute and relative errors, and error fluctuations are significantly controlled.

In Figure 4(b), on weekends, the overall error generally increases, but the PSO-TFT-GAT model still maintains the lowest error (MAE = 1.7 people/hour, RMSE = 1.3 people/hour, MAPE = 0.5%). The Informer-LSTM model achieved a MAE of 7.7 people/hour, an RMSE of 6.9 people/hour, and a MAPE of 3.2%, representing a MAE increase of 6.0 people/hour compared to the PSO-TFT-GAT model. The error of other models increases more significantly on weekends (for example, the MAE of the SAE-LSTM model increases from 15.0 people/h on weekdays to 18.9 people/h), indicating that weekend traffic poses a more stringent test on the robustness of the baseline model. However, the relative stability of the PSO-TFT-GAT model is more prominent on weekends.

In Figure 4(c), during statutory holidays, the errors of each model increase further, which are extreme peak traffic scenarios. The error of the PSO-TFT-GAT model also increases slightly, but it still performs optimally (MAE = 2.6 people/h, RMSE = 2.4 people/h, MAPE = 1.0 people/h). Compared to the Informer-LSTM (MAE = 8.1 people/h, RMSE = 7.0 people/h, MAPE = 2.7%), the PSO-TFT-GAT model reduced its MAE by 5.5 people/h. While the absolute

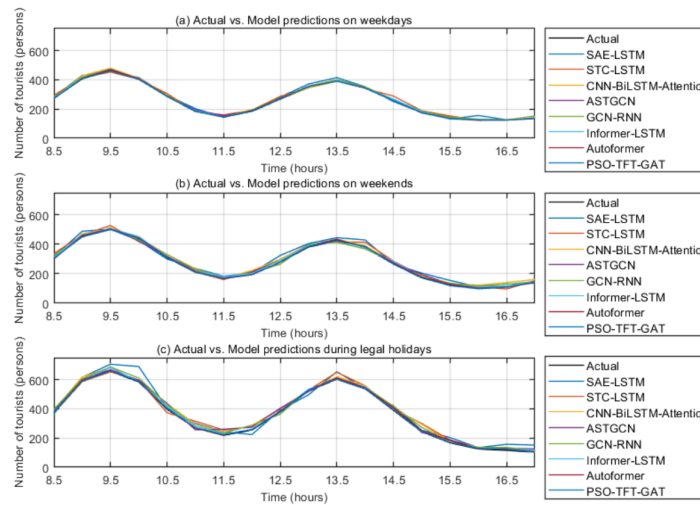


FIGURE 3. Comparison of Actual and Predicted Passenger Flow Curves from Different Models: (a) Weekdays; (b) Weekends; (c) Statutory Holidays

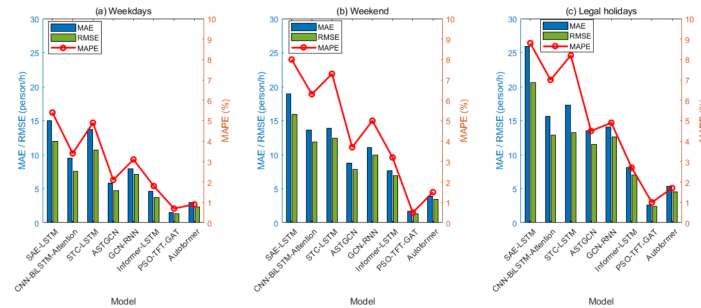


FIGURE 4. Overall Prediction Accuracy: (a) Weekdays; (b) Weekends; (c) Legal Holidays

error margins of all models increased during holidays (for example, the SAE-LSTM MAE reached 25.9 people/h), the PSO-TFT-GAT model maintained a smaller error increase compared to the baseline, demonstrating its superior control over extreme fluctuations compared to other methods. The experiment takes statutory holiday samples and runs each model $n = 30$ times (30 independent random seeds) under the same data segmentation and the same random seed set, and records the RMSE running in the holiday scenario. The steps are as follows:

- (1) Calculate the RMSE difference between the comparison model and the PSO-TFT-GAT model in this paper, and the sample mean and standard deviation of the difference.
- (2) Calculate the t statistic with a significance level of 0.05 and a 95% confidence interval
- (3) Null hypothesis H_0 : There is no difference in the RMSE of the two models during holidays.
- (4) Alternative hypothesis H_1 : There is a significant difference in the RMSE of the two models during holidays. The t test results are shown in Table 2.

The t -test results in Table 2 show that the RMSE differences between the PSO-TFT-GAT model and all comparison models in the statutory holiday scenario are highly significant (p -values are all far less than 0.05), rejecting the

null hypothesis H_0 . The PSO-TFT-GAT model's holiday RMSE is 2.4 ± 0.35 people/hour, while the SAE-LSTM, CNN-BiLSTM-Attention, and STC-LSTM models are 20.6 ± 2.6 people/hour, 12.9 ± 1.5 people/hour, and 13.3 ± 1.7 people/hour, respectively, all significantly higher than the PSO-TFT-GA. The RMSEs of the ASTGCN, GCN-RNN, Informer-LSTM, and Autoformer models were 11.5 ± 1.4 people/h, 12.6 ± 1.6 people/h, 7.0 ± 0.9 people/h, and 4.5 ± 0.6 people/h, respectively, all significantly higher than those of the PSO-TFT-GAT model. The 95% confidence intervals for these differences ranged from [1.80, 2.40] people/h (Autoformer) to [17.08, 19.32] people/h (SAE-LSTM). The t -values were all greater than 14, with the highest reaching 33.2, further demonstrating that the mean differences were statistically significant.

QUANTILE EVALUATION

This paper evaluates the quantile performance of each model in three scenarios (weekdays, weekends, and statutory holidays). The metrics include PICP (Prediction Interval Coverage Probability) (%), PINW (Prediction Interval Normalized Width) (the average width of the prediction interval between 10% and 90%), and CRPS (Continuous Ranked Probability Score). The results are shown in Figure 5.

TABLE 2. t test results

Comparison model	Model in this paper: PSO-TFT-GAT	Comparison model RMSE (mean $\pm$ std)	t-value (df=29)	p-value	95% Confidence interval (difference, person/hour)
SAE-LSTM	2.40 $\pm$ 0.35	20.6 $\pm$ 2.6	33.2	< 1e-25	[17.08, 19.32]
CNN-BiLSTM-Attention	2.40 $\pm$ 0.35	12.9 $\pm$ 1.5	30.3	< 1e-24	[9.79, 11.21]
STC-LSTM	2.40 $\pm$ 0.35	13.3 $\pm$ 1.7	28.5	< 1e-23	[10.12, 11.68]
ASTGCN	2.40 $\pm$ 0.35	11.5 $\pm$ 1.4	27.7	< 1e-23	[8.43, 9.77]
GCN-RNN	2.40 $\pm$ 0.35	12.6 $\pm$ 1.6	27.9	< 1e-23	[9.45, 10.95]
Informer-LSTM	2.40 $\pm$ 0.35	7.0 $\pm$ 0.9	22.9	< 1e-20	[4.19, 5.01]
Autoformer	2.40 $\pm$ 0.35	4.5 $\pm$ 0.6	14.4	< 1e-14	[1.80, 2.40]

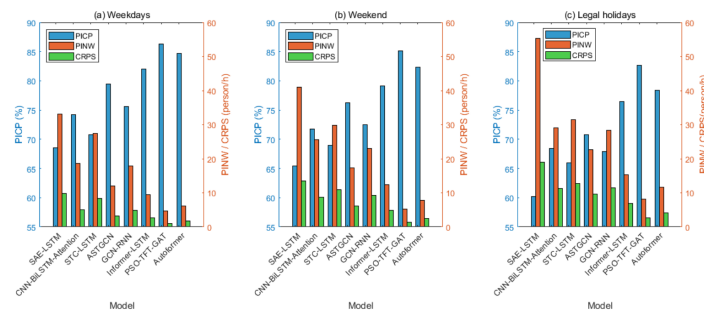


FIGURE 5. Percentile Evaluation: (a) Weekdays; (b) Weekends; (c) Holidays

The PSO-TFT-GAT model demonstrates superior probabilistic forecasting across all scenarios compared to the Informer-LSTM baseline, showcasing better calibration and sharpness.

On weekdays, the PSO-TFT-GAT achieves the best performance with a PICP of 86.3% and a narrow PINW of 4.7 people/h, significantly improving coverage (4.3%) and sharpness (≈ 4.7 people/h) over the baseline.

Weekends introduce higher volatility, causing a slight increase in the model's PINW and CRPS, but the PSO-TFT-GAT still maintains a strong PICP of 85.1% and a clear advantage, reducing the baseline's interval width by 7.2 people/h.

Figure 6 shows the quantile loss results for each model in three scenarios (weekdays, weekends, and statutory holidays), with the pinball loss at the 0.1, 0.5, and 0.9 quantiles (unit: person/hour).

In Figure 6 (a), on weekdays, all models achieve generally low Pinball losses at the 0.1/0.5/0.9 quantiles, with the PSO-TFT-GAT model significantly outperforming other methods, achieving 0.5/0.9/1.3 people/hour, respectively. The Informer-LSTM (1.6/2.2/3.4 people/hour) serves as a reference. The PSO-TFT-GAT model achieves absolute reductions of 1.1/1.3/2.1 people/hour at these quantiles. Compared to traditional deep models (e.g., SAE-LSTM's 8.3/9.1/10.2 people/hour), the improvements are even more significant, demonstrating that proposed model can produce sharper and better-calibrated prediction distributions in stable, highly cyclical weekday scenarios. Figure 6(b) shows that overall uncertainty increases on weekends, but the PSO-TFT-GAT model still maintains the lowest Pinball (0.6, 1.0, and 1.6 people/h), while the Informer-LSTM model achieves 3.0, 4.1, and 5.8 people/h. The absolute decreases are 2.4, 3.1, and 4.2 people/h, respectively. Other models (such as ASTGCN) experience larger increases in

Pinball on weekends, indicating that when intraday traffic fluctuations and sudden behaviors increase, the baseline model's uncertainty representation is significantly insufficient. The PSO-TFT-GAT model maintains a narrow range while maintaining a small bias.

Figure 6(c) shows that uncertainty is significantly amplified during statutory holidays, with Pinball values for all models generally increasing: 1.9, 2.6, and 3.5 people/hour for the PSO-TFT-GAT model, and 4.0, 5.6, and 7.4 people/hour for the Informer-LSTM model. The PSO-TFT-GAT model maintains its relative advantage, with absolute decreases of 2.1, 3.0, and 3.9 people/hour. However, the Pinball value of the PSO-TFT-GAT model increases significantly during holidays compared to weekdays and weekends (for example, the 0.1 percentile increases from 0.5 to 0.6 to 1.9), indicating that extreme peaks and heavy-tailed events can still cause forecast uncertainty to increase dramatically.

VISUALIZATION OF THE ATTENTION WEIGHT MATRIX AND ANALYSIS OF THE IMPORTANCE OF DIFFERENT METRICS

To further reveal the model's focus when predicting passenger flow, this study visualized the attention weight matrix and calculated different input indicators, including the importance of exogenous factors, based on the feature dimensions. The results are shown in Figure 7.

The model's spatial attention distribution (Figure 7a) reveals a hierarchical structure prioritizing key visitor flow paths. Core channel connections, such as Taihemen to Baohedian (weights up to 0.92), show high attention, as do interactions between central and surrounding areas (e.g., Jiaotaidian and Kunming Palace, 0.76–0.86). Significant gateway influence is also captured, with high weights between gates (Wumen, Donghuamen) and core attractions (0.45–0.95).

Indicator importance (Figure 7b) confirms that historical

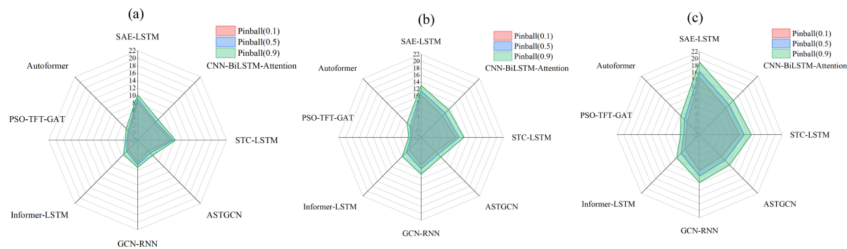


FIGURE 6. Percentile Loss Evaluation: (a) Weekdays; (b) Weekends; (c) Holidays

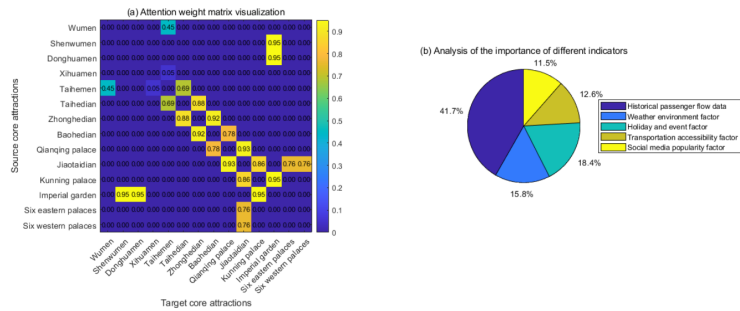


FIGURE 7. Visualization of the Attention Weight Matrix and Analysis of the Importance of Different Metrics: (a) Visualization of the Attention Weight Matrix; (b) Analysis of the Importance of Different Metrics

passenger flow is the primary driver (41.7%), followed by crucial exogenous factors like holidays/events (18.4%) and weather (15.8%). The consistency between attention weights and factor proportions verifies the TFT-GAT model’s effectiveness in capturing both core temporal dependencies and integrating multi-source spatial information.

SENSITIVITY OF DIFFERENT FORECAST STEPS

In order to explore the sensitivity of different prediction step sizes, this paper uses RMSE as an indicator to explore the curve of prediction accuracy changing with prediction step size. The results are shown in Figure 8.

Figures 8 (a)–(c) show that the prediction accuracy of each model decreases to varying degrees as the prediction step size increases. For example, on weekdays, the 1-step RMSE of the PSO-TFT-GAT model is 0.8 people/hour, while the 24-step RMSE increases to 2.6 people/hour, showing the smallest increase. The 1- step RMSE of the SAE-LSTM model is 6 people/hour, and the 24-step RMSE reaches 19 people/hour, a significantly greater increase. Similar trends are observed on weekends and weekdays. On weekends, the 1- step RMSE of the PSO-TFT-GAT model is 0.9 people/hour and the 24-step RMSE is 2.5 people/hour. On statutory holidays, the 1-step RMSE of the PSO-TFT-GAT model is 1.5 people/hour and the 24-step RMSE is 4.8 people/hour. These performances are superior to other comparison models. For example, the 24-step RMSE of the Informer-LSTM model is 11.5 people/hour and the Autoformer model is 7.5 people/hour on statutory holidays.

SCHEDULING PERFORMANCE ANALYSIS

To compare the performance of different scheduling methods in a realistic simulation environment, the paper used four indicators: average waiting time, traffic flow, congestion duration, and operating costs. The results are shown in Table 3.

TABLE 3. Scheduling Performance Analysis

Methods	Average waiting time	Traffic volume	Congestion duration	Operating cost
D3QN	4.2	320.5	1.2	1250.7
DQN	6.8	295.3	2.4	1100.4
A2C	5.1	305.7	1.8	1302.9
Rainbow DQN	4.9	310.2	1.6	1350.2
Rule-based scheduling	9.3	260.8	3.7	900.6

As shown in Table 3, the D3QN-based scheduling strategy performs best in all four metrics. Its average waiting time is 4.2 minutes, significantly lower than DQN’s 6.8 minutes and the rule-based scheduling’s 9.3 minutes. Its throughput rate reaches 320.5 people/hour, higher than DQN’s 295.3 people/hour and the rule-based scheduling’s 260.8 people/hour. Its congestion duration is only 1.2 hours per day, significantly less than DQN’s 2.4 hours and the rule-based scheduling’s 3.7 hours. The operating cost is 1250.7 yuan/day. Taking into account both service efficiency and cost control, this outperforms Rainbow DQN (Rainbow Deep Q- Network) at 1350.2 yuan/day and A2C (Advantage Actor Critic) at 1302.9 yuan/day.

REAL-TIME PERFORMANCE ANALYSIS

To evaluate the real-time performance of each prediction and scheduling solution, this article analyzed the model’s end-

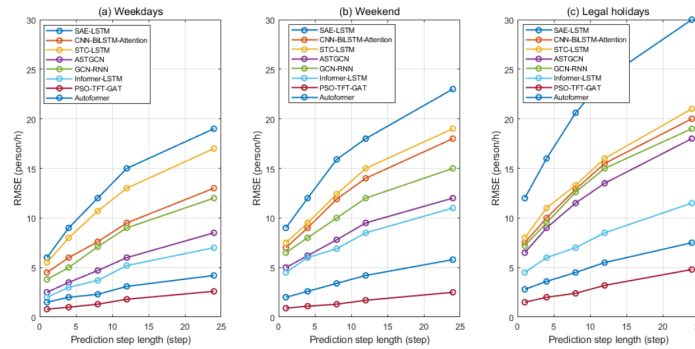


FIGURE 8. Prediction Accuracy Variation with Prediction Step Size: (a) Weekdays; (b) Weekends; (c) Legal Holidays

to-end training time, single-step inference latency, decision response time, model parameter count, and memory usage. The results are shown in Table 4.

Table 4 shows significant differences in real-time performance among the various prediction and scheduling solutions. Taking end-to-end training as an example, PSO-TFT-GAT (D3QN) trains in 9.5 hours, demonstrating higher training efficiency compared to ASTGCN (D3QN) (14.6 hours) and Autoformer (D3QN) (18.2 hours). Single-step inference latency is 18.7 ms, lower than ASTGCN (25.3 ms) and Autoformer (28.0 ms), but slightly higher than STC-LSTM (D3QN) (12.5 ms). The response time was 140.6 ms, a 34.8 ms reduction compared to the Informer-LSTM (D3QN) approach. While meeting the real-time requirements for scenic spot scheduling, the comprehensive model's parameter count of 8.9M and memory usage of 620.1MB are both moderate, significantly outperforming the Autoformer (18.7M, 910.3MB), which has a larger parameter count and higher memory usage.

GENERALIZATION AND ROBUSTNESS EVALUATION

To test the generalization and robustness of proposed method across diverse scenic areas, extreme weather conditions, and emergencies, the paper conducted a batch evaluation on several representative scenarios using a high-fidelity simulator. The results are shown in Figure 9. The various tourist attractions included Shanghai Disneyland, Hangzhou West Lake Scenic Area, Nanjing Sun Yat-sen Mausoleum, Beijing Summer Palace, and Guangzhou Chimelong Park. Extreme weather conditions included heavy rain, strong winds, heavy snow, high temperatures, and smog. Emergencies included celebrity visits, temporary exhibitions, stampedes, temporary security checkpoint closures, and fires.

The PSO-TFT-GAT algorithm demonstrated strong generalization capability across various tourist attractions (Figure 9a). While the Beijing Forbidden City showed the lowest error (MAE 1.5, RMSE 1.3 people/h), other sites also exhibited high accuracy. The Beijing Summer Palace achieved the best transfer result (MAE 1.7, RMSE 2.0), with errors slightly higher at Hangzhou West Lake (MAE 2.1) and large theme parks like Shanghai Disneyland (MAE 3.2) and Guangzhou Chimelong (MAE 3.8). The increased error in

large parks is attributed to more complex, fluctuating flow patterns and diverse visitor types compared to the highly regular flow at the Forbidden City.

The model showed robustness to extreme weather (Figure 9b), though prediction errors correlate with severity. Errors ranged from MAE 3.9 (high temperature) up to MAE 6.2 (heavy snow), confirming that extreme conditions introduce uncertainty but forecasts remain acceptable.

ABLATION EXPERIMENTS

To quantify the contribution of each module to overall performance, the paper designed ablation experiments by gradually removing each structural unit and comparing the changes in prediction accuracy and online response latency. All ablation versions were trained and evaluated repeatedly with the full model using the same data partitioning, training/inference hardware, and random seed to ensure comparability. The ablation results are shown in Table 5.

The steps are as follows: Full model: PSO + TFT + GAT + D3QN, with PSO removed and hyperparameters manually fine-tuned using the validation set; only TFT (without GAT and PSO) was retained, and decisionmaking was performed jointly with D3QN.

The ablation test results in Table 5 show that each module contributes significantly to overall performance. The full PSO + TFT-GAT (D3QN) model achieves a MAE and RMSE of 1.5 and 1.3 people/h, respectively, in weekday scenarios, with a response time of 140.6 ms. Removing the PSO module (relying on TFT-GAT and D3QN for joint decision-making), the MAE rises to 2.0 and RMSE to 2.5 people/h, while the response time decreases slightly to 136.8 ms. Further retaining only TFT (without GAT and PSO), the MAE and RMSE increased to 2.7 and 3.4 people/h, respectively, while the response time decreased to 122.4 ms.

TABLE 4. Real-time Performance Analysis Results

Model	Training time (h)	Single-step inference latency (ms)	Response time (ms)	Model parameters (M)	Memory usage (MB)
STC-LSTM(D3QN)	7.2	12.5	120.3	3.4	420.5
SAE-LSTM(D3QN)	6.8	10.9	110.4	2.7	380.2
CNN-BiLSTM-Attention(D3QN)	9.8	20.1	148.7	9.6	660.0
ASTGCN(D3QN)	14.6	25.3	170.5	15.4	820.6
GCN-RNN(D3QN)	11.3	22.4	155.0	12.2	710.8
Informer-LSTM(D3QN)	16.1	26.8	175.4	14.1	840.9
PSO-TFT-GAT(D3QN)	9.5	18.7	140.6	8.9	620.1
Autoformer(D3QN)	18.2	28.0	180.2	18.7	910.3

TABLE 5. Ablation Results

Complete model	TFT	GAT	PSO	MAE(person/h)	RMSE(person/h)	Response time (ms)
Full model	✓	✓	✓	1.5	1.3	140.6
PSO+TFT-GAT(D3QN)	✓	✓	-	2.0	2.5	136.8
TFT(D3QN)	✓	-	-	2.7	3.4	122.4

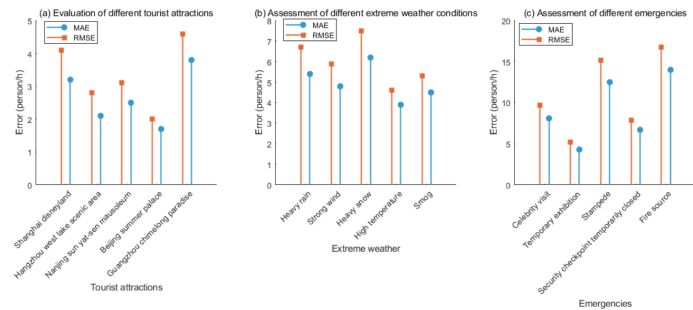


FIGURE 9. Generalization and Robustness Analysis Results: (a) Evaluation of different tourist attractions; (b) Evaluation of different extreme weather conditions; (c) Evaluation of different emergencies

VI. CONCLUSION

This paper introduces a Transformer-based integrated forecasting and scheduling framework designed to manage complex systems by fusing multi-source heterogeneous data. The architecture utilizes a Graph Attention Network (GAT) to model time-varying spatial relationships and integrates the Temporal Fusion Transformer (TFT) to achieve high-precision time series forecasting, with key hyperparameters optimized via Particle Swarm Optimization (PSO). This approach offers a powerful methodology directly applicable to the supply chain, where complex logistical and production decisions require accurate upstream forecasts. Consequently, the model’s prediction results and quantile information are input into a Deep Double Q- Network (D3QN) scheduling module to achieve a closed-loop linkage between forecasting and scheduling, ensuring dynamic, optimized resource deployment in environments characterized by high volatility, such as tourist flow or fashion demand. Experimental results show that, in the Beijing Forbidden City scenario, the prediction MAE for weekdays, weekends, and holidays is 1.5, 1.7, and 2.6 people/hour, respectively, with quantile coverage close to 82% to 86%. The MAE for prediction on statutory holidays is 13 people/hour lower than that of the traditional CNN-BiLSTM-Attention model. The average waiting time in the dispatch phase is shortened to 4.2 minutes, and the response time is approximately 140.6 milliseconds, significantly outperforming traditional rule-based methods. This framework significantly improves visitor flow prediction accuracy and dispatch responsiveness.

Despite its efficacy, limitations persist concerning cross-site adaptability (generalizing to new locations/products) and the robust modeling of complex human behaviors (tourist or consumer). These challenges, especially generalizing across diverse operational contexts and modeling heterogeneous decision-making, are crucial avenues for future development in both tourism supply chain optimization.

AUTHOR CONTRIBUTIONS

Conceptualization – Xiaona Xie and Wenliang Zhang; methodology – Xiaona Xie and Wenliang Zhang; investigation – Xiaona Xie and Wenliang Zhang; writing-original draft preparation – Xiaona Xie and Wenliang Zhang. All authors have read and agreed to the published version of the manuscript.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

FUNDING

This research received no external funding.

ACKNOWLEDGEMENTS

Not applicable.

REFERENCES

- [1] Y. Liu, G. Feng, K. S. Chin, S. Sun, and S. Wang, "Daily tourism demand forecasting: The impact of complex seasonal patterns and holiday effects," *Current Issues in Tourism*, vol. 26, no. 10, pp. 1573-1592, 2023, doi: 10.1080/13683500.2022.2060067.
- [2] A. Zvaigzne, L. Litavniece, and I. Dembovska, "Tourism seasonality: The causes and effects," *Worldwide Hospitality and Tourism Themes*, vol. 14, no. 5, pp. 421-430, 2022, doi: 10.1108/WHATT-07-2022-0080.
- [3] S. Wang and X. Yang, "Twice clustering based hybrid model for short-term passenger flow forecasting," *Transport*, vol. 39, no. 3, pp. 209-228, 2024, doi: 10.3846/transport.2024.20538.
- [4] S. He, "Tourist flow analysis and regulation strategy in tourist attractions based on spatiotemporal evolution and internet of things big data," *Journal of Biotech Research*, vol. 20, no. 1, pp. 89-97, 2025.
- [5] C. Munoz, A. Alvarez, and J. F. Banos, "Modelling the effect of weather on tourism: does it vary across seasons?," *Tourism Geographies*, vol. 25, no. 1, pp. 265-286, 2023, doi: 10.1080/14616688.2020.1868019.
- [6] J. Yu, "Passenger flow prediction of tourist attractions by integrating differential evolution and GWO," *Informatica*, vol. 48, no. 13, pp. 31-50, 2024, doi: 10.31449/inf.v48i13.6159.
- [7] Y. Liu, G. Feng, K. S. Chin, S. Sun, and S. Wang, "Daily tourism demand forecasting: The impact of complex seasonal patterns and holiday effects," *Current Issues in Tourism*, vol. 26, no. 10, pp. 1573-1592, 2023, doi: 10.1080/13683500.2022.2060067.
- [8] A. Zvaigzne, L. Litavniece, and I. Dembovska, "Tourism seasonality: The causes and effects," *Worldwide Hospitality and Tourism Themes*, vol. 14, no. 5, pp. 421-430, 2022, doi: 10.1108/WHATT-07-2022-0080.
- [9] S. Wang and X. Yang, "Twice clustering based hybrid model for short-term passenger flow forecasting," *Transport*, vol. 39, no. 3, pp. 209-228, 2024, doi: 10.3846/transport.2024.20538.
- [10] S. He, "Tourist flow analysis and regulation strategy in tourist attractions based on spatiotemporal evolution and internet of things big data," *Journal of Biotech Research*, vol. 20, no. 1, pp. 89-97, 2025.
- [11] C. Munoz, A. Alvarez, and J. F. Banos, "Modelling the effect of weather on tourism: does it vary across seasons?," *Tourism Geographies*, vol. 25, no. 1, pp. 265-286, 2023, doi: 10.1080/14616688.2020.1868019.
- [12] J. Yu, "Passenger flow prediction of tourist attractions by integrating differential evolution and GWO," *Informatica*, vol. 48, no. 13, pp. 31-50, 2024, doi: 10.31449/inf.v48i13.6159.
- [13] Q. Tang, L. Yang, and L. Pan, "Passenger flow forecast of tourist attraction based on MACBL in LBS big data environment," *Open Geosciences*, vol. 15, no. 1, pp. 1-12, 2023, doi: 10.1515/geo-2022-0577.
- [14] A. G. Vrahatis, K. Lazaros, and S. Kotsiantis, "Graph attention networks: A comprehensive review of methods and applications," *Future Internet*, vol. 16, no. 9, pp. 318-351, 2024, doi: 10.3390/fi16090318.
- [15] D. Yin, R. Jiang, J. Deng, Y. Li, Y. Xie, Z. Wang, et al., "MTMGNN: Multi-time multi-graph neural network for metro passenger flow prediction," *GeoInformatica*, vol. 27, no. 1, pp. 77-105, 2023, doi: 10.1007/s10707-022-00466-1.
- [16] L. Liu, Y. Liu, and X. Ye, "Multi-sequence spatio-temporal feature fusion network for peak-hour passenger flow prediction in urban rail transit," *Transportation Letters*, vol. 17, no. 1, pp. 86-102, 2025, doi: 10.1080/19427867.2024.2327805.
- [17] Q. Xue, W. Zhang, M. Ding, X. Yang, J. Wu, and Z. Gao, "Passenger flow forecasting approaches for urban rail transit: A survey," *International Journal of General Systems*, vol. 52, no. 8, pp. 947, 2023, doi: 10.1080/03081079.2023.2231133.
- [18] J. Sun, Z. Qiu, Y. Sun, and O. Simpson, "A Self-Correction Transformer Network for Traffic Flow Prediction Under Dynamic Spatio-Temporal Distributions," *IET Intelligent Transport Systems*, vol. 19, no. 1, pp. 1-15, 2025, doi: 10.1049/itr2.70044.
- [19] B. Wu, L. Wang, and R. Zeng Y, "Interpretable tourism demand forecasting with Two-Stage decomposition and Temporal fusion Transformers," *Journal of Systems Science and Complexity*, vol. 37, no. 6, pp. 2654-2679, 2024, doi: 10.1007/s11424-024-2307-0.
- [20] Y. Lai and Y. Wang, "Short-term passenger flow prediction for rail transit based on improved particle swarm optimization algorithm," *IET Intelligent Transport Systems*, vol. 17, no. 4, pp. 825-834, 2023, doi: 10.1049/itr2.12306.
- [21] L. Zhao, H. Jin, X. Zou, and X. Liu, "Short-Term Passenger Flow Forecasting for Rail Transit considering Chaos Theory and Improved EMD-PSO-LSTM-Combined Optimization," *Journal of Advanced Transportation*, vol. 2023, no. 1, pp. 1-16, 2023, doi: 10.1155/2023/9524966.
- [22] B. Jiang, J. Zhang, J. Fu, G. Ding, and Y. Zhang, "Research on Check-In Baggage Flow Prediction for Airport Departure Passengers Based on Improved PSO-BP Neural Network Combination Model," *Aerospace*, vol. 11, no. 11, pp. 953-973, 2024, doi: 10.3390/aerospace11110953.
- [23] S. Hu, M. Ji, Z. Chang, H. Wang, and X. Kong, "An improved particle swarm optimization algorithm based urban rail passenger flow prediction model: a case study in Beijing, China," *Digital Transportation and Safety*, vol. 4, no. 2, pp. 101-107, 2025, doi: 10.48130/dts-0025-0005.
- [24] Y. Fu, Y. Shen, and L. Tang, "A dynamic task allocation framework in mobile crowd sensing with D3QN," *Sensors*, vol. 23, no. 13, pp. 6088-7100, 2023, doi: 10.3390/s23136088.
- [25] M. Reyad, A. M. Sarhan, and M. Arafa, "A modified Adam algorithm for deep neural network optimization," *Neural Computing and Applications*, vol. 35, no. 23, pp. 17095-17112, 2023, doi: 10.1007/s00521-023-08568-z.
- [26] C. Zhang, Y. Shao, H. Sun, L. Xing, Q. Zhao, and L. Zhang, "The WuC-Adam algorithm based on joint improvement of Warmup and cosine annealing algorithms," *Mathematical Biosciences & Engineering*, vol. 21, no. 1, pp. 1270-1285, 2024, doi: 10.3934/mbe.2024054.
- [27] N. Xiao, X. Hu, X. Liu, and K. C. Toh, "Adam-family methods for non-smooth optimization with convergence guarantees," *Journal of Machine Learning Research*, vol. 25, no. 48, pp. 1-53, 2024, [Online]. Available: <http://jmlr.org/papers/v25/23-0576.html>.