Artificial Intelligence and Emerging Technologies
https://www.sciltp.com/journals/aiet

Article

A Spatiotemporal Coupled Prediction Method for Multi-Energy Flows in Transportation Hubs Based on H3 Grid Representation and Heterogeneous Graph Attention–Transformer Networks

Weiyi Jiang, Liyan Zhang, Ze Zhou *, Dongqi Zhao and Qihong Chen *

School of New Energy and Electrical Engineering, Wuhan University of Technology, Wuhan 430070, China
* Correspondence: zhouze@whut.edu.cn (Z.Z.); chenqh@whut.edu.cn (Q.C.)

How To Cite: Jiang, W.; Zhang, L.; Zhou, Z.; et al. A Spatiotemporal Coupled Prediction Method for Multi-Energy Flows in Transportation Hubs Based on H3 Grid Representation and Heterogeneous Graph Attention–Transformer Networks. *Artificial Intelligence and Emerging Technologies* **2026**, *3*(3), 12. https://doi.org/10.53941/aiet.2026.100012

Received: 20 July 2026
Revised: 21 August 2026
Accepted: 1 September 2026
Published: 16 September 2026

Abstract: Large transportation hubs such as ports are becoming high-density agglomerations of energy consumption, in which electricity, hydrogen, and thermal/cooling energy flows are increasingly coupled with traffic activity, so their coordinated and low-carbon operation has become an important concern for both intelligent transportation and integrated energy systems. However, the multi-source data of such hubs differ greatly in spatial granularity and physical meaning, and traffic and energy loads are jointly coupled across space and time, which makes conventional methods that forecast a single system unable to characterize the overall dynamics. To address this problem, this paper proposes a heterogeneous data fusion and spatiotemporal coupled prediction method for multi-energy flows in transportation hubs. A unified representation framework based on H3 hexagonal grids is first designed, which aligns multi-source data onto a common spatial index and fuses them into a standardized fourth-order tensor through area, length, and frequency spatial weights together with quantile truncation. On this basis, a hybrid model that couples a heterogeneous graph attention network (HetGAT) with a Transformer encoder is constructed, in which HetGAT models the heterogeneous spatial topology formed by four node types and five edge types, while the Transformer encoder extracts long-range temporal dependencies. An uncertainty-weighted dual-task head then jointly outputs traffic states and multi-energy loads. Experiments on a Ningbo-Zhoushan Port scenario built with the SUMO simulation platform show that the proposed method achieves R^2 values of 0.9763, 0.8088, 0.9644, and 0.9583 for traffic flow, electricity load, hydrogen consumption, and thermal/cooling load, respectively, while consistently outperforming single-system and homogeneous-graph baselines on the energy-side prediction tasks.

Keywords: multi-source heterogeneous data fusion; H3 hexagonal grid; heterogeneous graph attention network; Transformer; spatiotemporal coupling prediction

1. Introduction

Large-scale transportation hubs, such as ports, airports, and railway freight stations, serve as core nodes in modern logistics networks and have become high-density agglomerations of energy consumption and carbon emissions. With the proposal of China's strategic goals of achieving carbon peaking by 2030 and carbon neutrality by 2060, low-carbon transportation has become a key direction for promoting the sustainable development of transportation infrastructure [1]. In this context, the coupling relationships among electricity flows, hydrogen flows, and thermal/cooling energy flows within transportation hubs have been continuously strengthened, gradually forming a structurally complex and interdependent multi-energy flow network [2].

In large-scale transportation hubs, the operational states of multi-energy flow systems are jointly affected by multiple factors, including fluctuations in transportation demand, road network topology, vehicle interaction behaviors, and unexpected incidents [3]. As a result, traffic flows and energy loads exhibit pronounced uncertainty and strong nonlinear characteristics across both temporal and spatial dimensions. Variations in transportation activities not only directly influence energy consumption levels but may also trigger cross-system cascading effects through energy conversion and supply scheduling processes. Conventional prediction methods designed for a single transportation system or a single energy system are therefore insufficient for accurately characterizing the overall dynamic behavior of such systems. Consequently, the coordinated prediction of traffic flows and multi-energy flows has become a critical scientific problem that urgently needs to be addressed in the fields of intelligent transportation and integrated energy systems. Figure 1 illustrates the multi-energy flow prediction scenario considered in this study and its downstream decision-support applications.

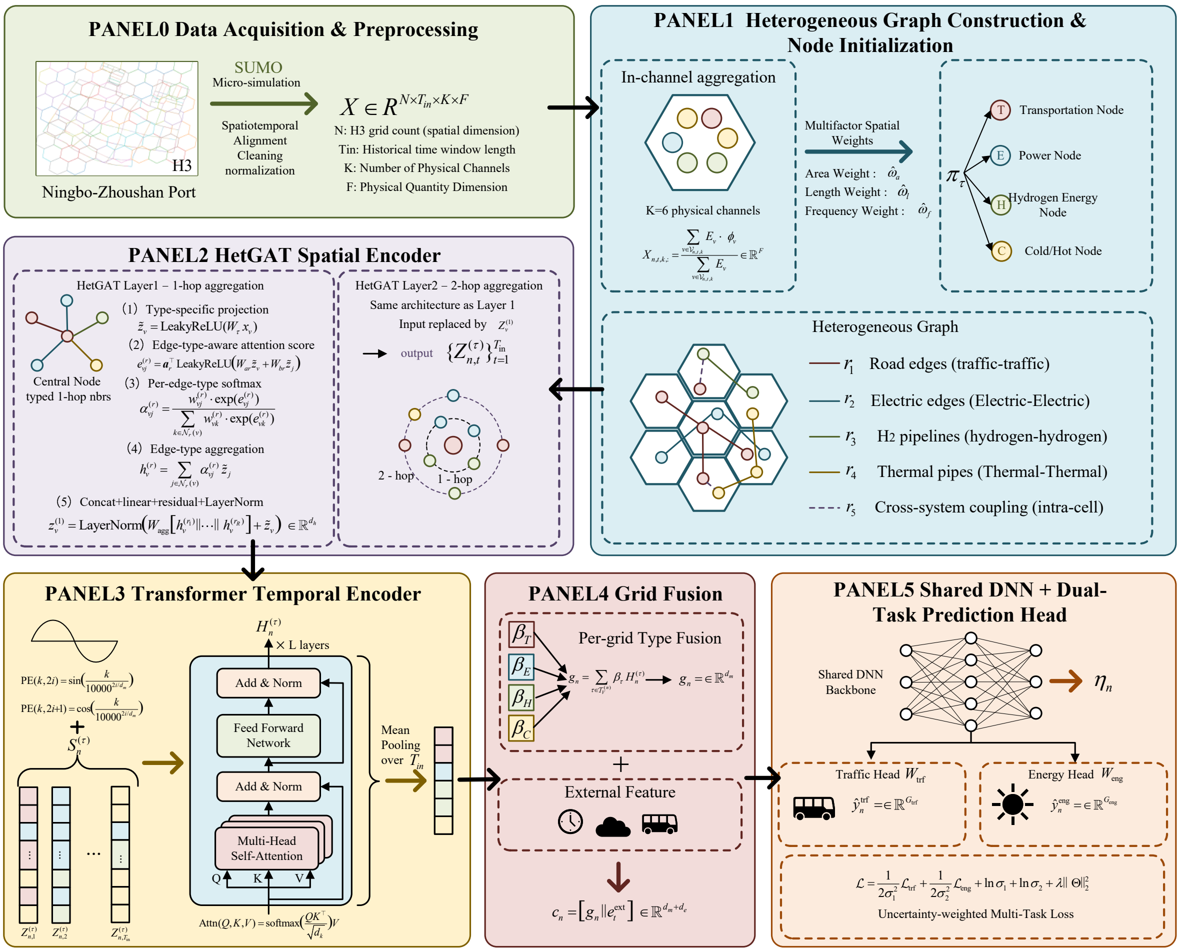

Figure 1. End-to-end data flow framework of the HetGAT–Transformer hybrid prediction model (high-level overview).

An Integrated Energy System (IES) enables the unified planning and optimized operation of multi-energy systems by exploiting the complementary characteristics of multiple energy carriers, such as electricity, heat, and gas [4]. The Energy Hub concept proposed by Geidl et al. provides a theoretical foundation for the collaborative modeling of multi-energy flows [5], while Mohammadi et al. further introduced a more comprehensive structure for multi-energy conversion and storage [6]. However, these methods mainly focus on static coupling and planning optimization. In large-scale transportation hubs, where transportation demand is a dominant driving factor and multiple energy carriers coexist, two major challenges remain. The first is the difficulty of achieving a unified representation of multi-source heterogeneous data. Traffic data, electricity, hydrogen and thermal energy monitoring data, and road network topology data are collected from different sources with inconsistent sampling granularities, which makes their fusion difficult. The second is the difficulty of jointly modeling the spatiotemporal coupling relationships, since the system contains both spatial topological dependencies among roads and energy facilities and temporal dependencies characterized by long-term trends, periodic patterns, and unexpected disturbances. A single model can usually capture only part of these complex dependencies.

https://doi.org/10.53941/aiet.2026.100012

In recent years, deep learning methods have achieved significant progress in spatiotemporal forecasting for integrated energy and transportation–energy systems. Recent studies on multi-energy load forecasting have increasingly explored the coupling among different energy carriers. Song et al. developed a hierarchical multi-task learning framework with spatiotemporal attention for the joint forecasting of cooling, heating, and electrical loads [7]. Su et al. introduced a spatio-temporal graph convolutional neural network to capture the correlations among multiple loads in regional integrated energy systems [8]. More recently, Kim et al. proposed a global-local attention-enabled multiple-decoder Transformer for multi-energy load forecasting, explicitly modeling coupling relationships and temporal dependencies [9]. In parallel, transportation–energy coupling has also attracted increasing attention. Yang et al. established a transportation–power coupled network and predicted the spatiotemporal distribution of electric-vehicle charging loads by incorporating mobility patterns, origin–destination information, and transportation conditions [10].

Graph neural networks (GNNs) can effectively characterize spatial dependencies over irregular topologies by aggregating information from neighboring nodes in graph structures [11]. Heterogeneous graph attention networks (HetGATs) further enable differentiated modeling of different types of nodes and edges [12]. Transformer models based on self-attention mechanisms have demonstrated strong capability in modeling long-range temporal dependencies [13]. Spatiotemporal joint frameworks that integrate graph neural networks with temporal modeling methods have therefore become an important research direction [14].

Despite these advances, existing multi-energy forecasting studies mainly focus on the coupling among energy loads within integrated energy systems, whereas transportation–energy studies are predominantly concerned with the interaction between traffic mobility and electric-vehicle charging demand. The joint prediction of traffic states together with electricity, hydrogen, and thermal/cooling energy flows over a unified heterogeneous spatial topology remains insufficiently explored.

To address these issues, this paper proposes a heterogeneous data fusion and spatiotemporal coupled prediction method for multi-energy flows in transportation hubs. The main contributions are summarized as follows:

(1) A unified representation method for multi-source heterogeneous data based on H3 hexagonal grids is proposed. The H3 spatial index is adopted to align multi-source data onto a unified spatial grid. Three types of spatial weights, corresponding to areal, linear, and point-based spatial elements, are designed to quantify the spatial coupling intensity among heterogeneous entities. Together with temporal alignment and quantile truncation normalization, a standardized four-dimensional tensor is constructed.

(2) A hybrid HetGAT–Transformer spatiotemporal encoding architecture is constructed to address the challenge of simultaneously modeling heterogeneous spatial topology and long-range temporal dependencies. HetGAT performs type-aware spatial message passing over four node types and five edge types, while the Transformer encoder captures long-range temporal dependencies. This architecture explicitly separates and subsequently integrates spatial heterogeneous encoding and temporal sequence modeling.

(3) An uncertainty-weighted multi-task joint learning strategy is developed to address cross-task coupling and optimization imbalance between traffic-state and energy-load forecasting. A shared backbone enables traffic and energy tasks to exchange coupled representations, while homoscedastic uncertainty weighting automatically balances their training objectives, thereby exploiting positive transfer between transportation activity and multi-energy demand.

The remainder of this paper is organized as follows. Section 2 describes the acquisition and preprocessing of the multi-source heterogeneous data based on the SUMO scenario of Ningbo-Zhoushan Port. Section 3 presents the unified data representation built on H3 hexagonal grids, including the fourth-order tensor and the three types of spatial weights. Section 4 details the proposed HetGAT–Transformer hybrid prediction model together with its uncertainty-weighted dual-task learning strategy. Section 5 reports the experimental setup, the ablation studies, and the comparison with baseline methods. Section 6 concludes the paper and outlines directions for future work.

2. Multi-Source Heterogeneous Data Acquisition and Preprocessing

2.1. Construction of the SUMO Simulation Scenario and Data Acquisition

This study employs the open-source microscopic traffic simulation platform SUMO (Simulation of Urban MObility) to construct the operational scenario of a transportation hub [15,16]. Meanwhile, synthetic Supervisory Control and Data Acquisition (SCADA) data of fixed port equipment are introduced to cover typical multi-energy flow operation scenarios at Zhoushan Port.

Traffic Mobile-Source Data. The simulation network is constructed based on the road network extracted from satellite maps of Ningbo-Zhoushan Port, thereby preserving the spatial layout and road-network topology of the actual port area. The number of vehicles operating in the network reaches the thousand-vehicle scale, and the vehicle types include fuel-powered vehicles, electric vehicles, and hydrogen-powered vehicles. Vehicle routes are generated using scripts within this real-world road network. Although the road-network topology is derived from the actual port map, the generated traffic demand and route-choice patterns are not explicitly calibrated against empirical origin–destination distributions or measured traffic counts from the port. Therefore, the routing process represents a simplification of actual port traffic operations while retaining the realistic spatial characteristics of the port road network. Floating car data (FCD) and emission output files generated by SUMO are used to obtain vehicle spatial locations, operating speeds, and energy consumption information at different time steps. These data are integrated using the unique vehicle identifier and timestamp as indices, thereby forming a joint dataset that contains both traffic operation states and energy consumption information.

Fixed-Equipment Data. As a pilot hydrogen-energy port in the coastal Yangtze River Delta region, Zhoushan Port contains several representative types of fixed energy equipment, including shore power piles for supplying berthed electric vessels, battery swapping stations for electric container trucks, refrigerated container yards with continuous electricity demand, hydrogen refueling stations for hydrogen-powered container trucks and vessels, marine hydrogen refueling stations with high-flow refueling outlets on the dock side, and Heating, Ventilation and Air Conditioning (HVAC) heating/cooling source stations for office and storage areas. The power and flow time-series data of these fixed facilities are synthetically generated based on rated equipment parameters and typical operational patterns at Zhoushan Port. A fixed-equipment dataset is then constructed using the equipment ID, corresponding grid ID, and timestamp as indices.

Finally, spatiotemporal index-based data fusion is performed to construct a unified operational dataset covering both mobile sources and fixed sources.

2.2. Spatiotemporal Alignment and Data Cleaning

To ensure the applicability of the proposed method to real-world data, this study systematically preprocesses the multi-source data according to the following procedure.

In the temporal dimension, a discrete time grid with a fixed interval of $\Delta t = 5\,\mathrm{min}$ is constructed. Nearest-neighbor matching is adopted to map the original observations to the corresponding time nodes, thereby aligning mobile-source trajectory data and fixed-equipment SCADA data on a unified temporal scale.

In the spatial dimension, the Cartesian coordinates used internally by SUMO are uniformly transformed into longitude and latitude representations under the WGS84 geographic coordinate system. The locations of fixed equipment are directly represented by the GPS coordinates of their installation sites.

For data quality control, the interquartile range method [17] is employed to identify abnormal observations, which are then treated as missing values. Features with a missing rate greater than 30% are removed, while the remaining missing values are imputed by linear interpolation. Since SUMO records the energy consumption of all vehicles uniformly in terms of electricity by default, the energy consumption of fuel-powered and hydrogen-powered vehicles needs to be converted accordingly:

$$V_{\mathrm{fuel}}(\mathrm{L}) = \frac{E(\mathrm{Wh})}{3300} \tag{1}$$

$$m_{\mathrm{H_2}}(\mathrm{kg}) = \frac{E(\mathrm{Wh})}{20000} \tag{2}$$

where V_{fuel} denotes gasoline consumption, $m_{\mathrm{H_2}}$ represents hydrogen consumption, and E is the equivalent electricity consumption output by SUMO. The conversion coefficients account for the effective energy-conversion efficiency of the corresponding vehicle powertrains. For gasoline, an energy content of approximately 8.9 $\mathrm{kWh/L}$ together with an assumed effective gasoline-engine conversion efficiency of about 37% gives an equivalent useful energy of approximately 3.3 $\mathrm{kWh/L}$. For hydrogen, the lower heating value is approximately 33.3 $\mathrm{kWh/kg}$, and an assumed PEM fuel-cell efficiency of 60% gives an equivalent useful energy of approximately 20.0 $\mathrm{kWh/kg}$. These values lead to the conversion coefficients used in Equations (1) and (2).

3. Unified Representation of Multi-Source Heterogeneous Data

This section constructs a complete representation pipeline from the original sample pool to standardized spatiotemporal tensors. It mainly addresses two issues. The first is how to use a structurally explicit tensor to simultaneously represent mobile-source and fixed-source data, and the second is how to properly employ three types of spatial weights while avoiding dimensional inconsistency.

3.1. Spatial Indexing Based on H3 Hexagonal Grids

The Uber H3 hexagonal grid indexing system is adopted as the unified spatial reference framework [18]. Compared with rectangular grids, hexagonal grids offer advantages in terms of equidistance, compactness, and adjacency consistency. In this study, resolution $r = 8$ is selected, corresponding to an average edge length of approximately $460\,\mathrm{m}$ and an average cell area of about $0.74\,\mathrm{km}^2$. In this study, resolution $r = 8$ is selected as a compromise between spatial representation accuracy and data sparsity. A sub-kilometer grid scale is sufficiently fine to preserve local differences in road-network structure and the spatial distribution of energy facilities, while avoiding excessive fragmentation into sparsely populated or empty cells. This scale is therefore suitable for representing local traffic–energy interactions within the port scenario. It should be emphasized that the proposed framework is not restricted to $r = 8$; for other transportation hubs, the H3 resolution can be adjusted according to road spacing, energy-facility density, data sampling density, and the desired computational granularity.

Each valid GPS observation point $P = (lon, lat, t)$ is mapped to a unique 64-bit index through H3 encoding:

$$h = H3Encode(lon, lat, r) \tag{3}$$

where h denotes the H3 index value, $H3Encode(\cdot)$ represents the H3 encoding function, lon and lat denote longitude and latitude, respectively, and r is the H3 resolution level. Figure 2 shows a schematic diagram of the H3 hexagonal grid mapping of Ningbo-Zhoushan Port.

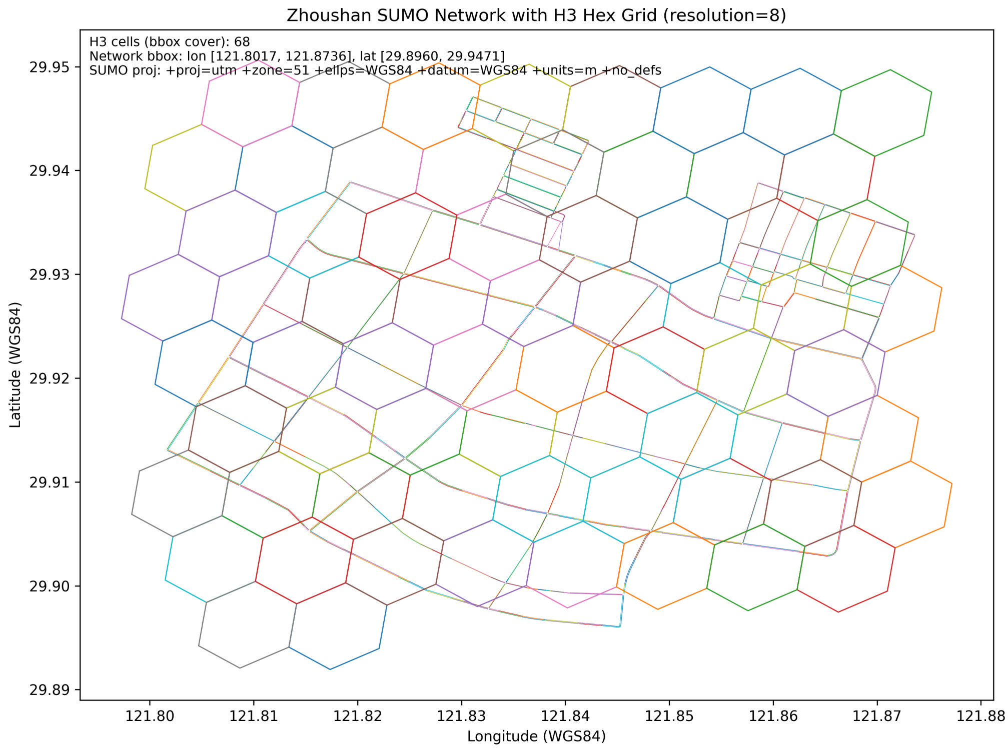

Figure 2. Schematic diagram of H3 hexagonal grid mapping of Ningbo-Zhoushan Port.

3.2. Fourth-Order Tensor Representation Based on Fixed Physical Channels

3.2.1. Definition of the Tensor Structure

The structures of the input tensor $\mathbf{X}$ and output tensor $\mathbf{Y}$ are defined as follows:

$$\mathbf{X} \in \mathbf{R}^{N \times T_{in} \times K \times F}, \quad \mathbf{Y} \in \mathbf{R}^{N \times T_{out} \times G} \tag{4}$$

where N denotes the number of H3 grids (spatial dimension), T_{in} represents the length of the historical observation window, K is the number of physical channels (fixed at $K = 6$ in this study), F denotes the dimensionality of physical variables within each channel, T_{out} is the number of forecasting time steps, and G represents the dimensionality of the prediction targets.

Here, the term "fourth-order tensor" refers to the number of indexing axes, namely the order of the tensor, rather than the length of any specific axis. The tensor $\mathbf{X}$ contains four mutually independent indexing axes, namely

N, T_{in}, K, and F. It is therefore referred to as a fourth-order tensor, or equivalently a four-dimensional tensor. In deep learning terminology, an "n-dimensional tensor" usually denotes an n-order tensor. The length of each axis may itself be large. Thus, "four-order" does not imply that the total dimensionality is only 4.

3.2.2. Definition of the K Dimension: Fixed Physical Channels

In this study, K is defined as the number of physical channels and is fixed at $K = 6$. Each channel is assigned a clear physical meaning, as shown in Table 1:

Table 1. Definitions of the six physical channels.

No.	Channel Name	Typical Scenarios at Zhoushan Port
1	Fuel-powered vehicles (mobile)	Passenger cars and fuel-powered freight trucks
2	Electric vehicles (mobile)	Electric container trucks and electric port machinery
3	Hydrogen-powered vehicles (mobile)	Hydrogen-powered container trucks and hydrogen fuel cell vehicles
4	Fixed electrical equipment	Shore power piles and battery swapping stations
5	Fixed hydrogen energy equipment	Hydrogen refueling stations for vehicles and vessels
6	Fixed heating/cooling equipment	HVAC heating/cooling source stations and cold-chain storage facilities

The six channels naturally correspond to the physical sources of the four target flows. Specifically, traffic flow is derived from the speed and flow information of Channels 1–3. Electricity flow is obtained from Channel 2 (mobile electricity consumption) and Channel 4 (fixed electrical equipment). Hydrogen flow is derived from Channel 3 (mobile hydrogen consumption) and Channel 5 (fixed hydrogen energy equipment), and thermal/cooling energy flow is represented by Channel 6.

3.2.3. Definition of the F Dimension: Physical Variables in Each Channel

Each channel records $F = 4$ physical variables:

$$\begin{aligned}\phi = \big(&\text{Traffic flow / operational intensity},\ \text{Electricity consumption rate [Wh/s]},\\ &\text{Hydrogen consumption rate [g/s]},\ \text{thermal}-\text{cooling consumption rate [kW]}\big)\end{aligned} \tag{5}$$

The first dimension (traffic flow/operational intensity) records the number of vehicles passing through a given grid within a unit time step for mobile channels (Channels 1–3), expressed as vehicles per time step. For fixed-equipment channels, it represents an operational intensity indicator of the corresponding facility, such as the number of containers handled per hour by a quay crane. A value of zero is assigned when no relevant operation exists. The second dimension (electricity consumption rate) and the third dimension (hydrogen consumption rate) record the electricity and hydrogen consumption rates, respectively, of vehicles or equipment within each channel. Channels that do not involve the corresponding energy carrier are assigned a value of zero. The fourth dimension (thermal/cooling consumption rate) is applicable only to the fixed thermal/cooling equipment channel (Channel 6), where it records the real-time thermal or cooling power (kW) of HVAC energy stations or cold-chain storage facilities. All mobile channels and the remaining fixed-equipment channels are assigned a value of zero for this dimension.

3.2.4. Aggregation Formula Within Each Channel

For each triplet (n, t, k), the channel feature values are obtained as the weighted average of the energy-related attributes of all original records belonging to that channel.

$$X_{n,t,k,:} = \frac{\sum\limits_{v \in \mathcal{V}_{n,t,k}} E_v \cdot \phi_v}{\sum\limits_{v \in \mathcal{V}_{n,t,k}} E_v} \in \mathbf{R}^F \tag{6}$$

where $\mathcal{V}_{n,t,k}$ denotes the set of all original records (vehicles or equipment) that belong to channel k and are located in grid n at time step t. The variable E_v represents the equivalent energy consumption (Wh) of record v and is used as the weighting factor, under the assumption that records with higher energy consumption are more representative of the operational state of the corresponding channel at that time. The vector ϕ_v denotes the F-dimensional feature vector associated with record v.

If $\mathcal{V}_{n,t,k} = \emptyset$, indicating that no record exists for channel k in grid n at time step t, then $X_{n,t,k,:} = \mathbf{0}$ is assigned and an additional mask indicator $m_{n,t,k} = 0$ is introduced. This mask allows the subsequent model to automatically ignore empty channels during training and inference.

3.3. Separate Use of Spatial Weights for Multi-Source Elements

When multi-source heterogeneous data are mapped into the same H3 grid, their spatial coupling intensity needs to be quantified. For three types of spatial elements, namely areal, linear, and point-based elements, area weights, length weights, and frequency weights are defined, respectively.

3.3.1. Area Weight $\hat{\omega}_a$: Static Infrastructure Profile of Nodes

For each grid n, the facility coverage ratio is calculated separately according to the facility type $\tau \in traffic, electricity, hydrogen, heating/cooling$:

$$\hat{\omega}_a^{(\tau)}(n) = \frac{\sum_e S_{\text{inter}}(e,n) \cdot \mathbf{1}[\text{type}(e) = \tau]}{S_{\text{grid}}} \tag{7}$$

where $\hat{\omega}_a^{(\tau)}(n) \in [0,1]$ denotes the normalized area weight of type-τ facilities within grid n. The hat symbol indicates normalization, the subscript a denotes area, and the superscript $\tau \in \mathcal{T}_V = T, E, H, C$ represents the node type. Here, e denotes a geographic element within the grid, such as a building, substation, hydrogen refueling station, or other physical entity. $S_{\text{inter}}(e,n)$ denotes the intersection area between element e and grid n. The term $\mathbf{1}[\text{type}(e) = \tau]$ is an indicator function that takes the value of 1 if the type of element e is τ and 0 otherwise, thereby serving to filter spatial elements by type. S_{grid} denotes the total area of a single H3 grid.

This procedure yields a four-dimensional vector $\hat{\omega}_a(n) \in \mathbf{R}^4$, which highlights the importance of the grid with respect to different infrastructure categories. This vector is subsequently concatenated with the initial feature vector of each node to provide static spatial priors.

3.3.2. Length Weight $\hat{\omega}_l$: Prior Weight of Edges

For linear infrastructure of type r between adjacent grids (n, n'), the normalized length is defined as follows:

$$w_{nn'}^{(r)} = \hat{\omega}_l^{(r)}(n,n') = \frac{l_{\text{inter}}^{(r)}(n,n')}{l_{\text{ref}}^{(r)}} \in [0,1] \tag{8}$$

where $w_{nn'}^{(r)} = \hat{\omega}_l^{(r)}(n,n') \in [0,1]$ denotes the normalized length weight of an edge of type r connecting grids n and n'. The hat symbol indicates normalization, the subscript l denotes length, and the superscript $r \in \mathcal{T}_E$ identifies the edge type. Here, $l_{\text{inter}}^{(r)}(n,n')$ represents the intersection length of a type-r linear feature, such as a road, power transmission line, or hydrogen pipeline, between grids n and n'. This quantity characterizes the physical connectivity strength between the two grids along the corresponding channel. A larger intersection length implies a stronger connection. The term $l_{\text{ref}}^{(r)}$ denotes the reference length for type-r edges and serves as the normalization factor, transforming the intersection length into a relative strength within the range $[0,1]$.

This value is directly used as the prior weight of edge (n, n', r) in the heterogeneous graph and is incorporated as a multiplicative bias in the softmax normalization of the HetGAT attention mechanism. Intuitively, the longer the road, pipeline, or transmission line traversing two grids, the greater the capacity for material or energy transfer between the corresponding nodes, and thus the larger the edge weight.

3.3.3. Frequency Weight $\hat{\omega}_f$: Dynamic Time-Series Feature of Event Intensity

Events, such as traffic congestion alerts, vessel berthing/departure operations, and battery-swapping completions, are essentially timestamped point processes. Therefore, they should not be compressed into a single static scalar. Instead, their temporal dynamics should be preserved:

$$\hat{\omega}_f(n,t) = \text{number of events occurring in grid } n \text{ within the time window } [t - \Delta t,\, t] \tag{9}$$

This value is used as a dynamic time-series feature and is concatenated with the channel data into the initial node features, enabling the model to perceive real-time variations in event activity within a given grid at a specific time.

3.4. Temporal Alignment and Feature Normalization

A global time index $t_k = t_0 + k \cdot \Delta t$ is generated using a common sampling interval of $\Delta t = 5\ \mathrm{min}$, and all data are aligned onto an equally spaced discrete time axis.

For continuous features, quantile truncation and Z-score normalization are jointly applied. Specifically, the 1st and 99th percentiles, denoted as q_1 and q_{99}, are first calculated to define the valid value range. The features are then clipped to this interval and standardized as follows:

$$z = \frac{\mathrm{clip}(x,\, q_1,\, q_{99}) - \mu}{\sigma} \tag{10}$$

where μ and σ denote the mean and standard deviation computed over the historical window of no less than 21 days. This two-step procedure effectively suppresses the interference of extreme values while eliminating dimensional discrepancies among features. For empty channels corresponding to the mask indicator $m_{n,t,k} = 0$, the feature values are directly set to zero and are excluded from the normalization statistics.

3.5. From Channel Tensors to Initial Node Features

After tensor construction and spatial weight calculation are completed, the initial node feature vector for each grid n at time step t is constructed by directly concatenating the corresponding physical channels according to the node type τ:

$$\begin{aligned} x_{n,t}^{(T)} = W_T \cdot \mathrm{concat}\big(&X_{n,t,1,:},\ X_{n,t,2,:},\ X_{n,t,3,:}, \\ &\hat{\omega}_a(n),\ \hat{\omega}_f(n,t)\big) + b_T \end{aligned} \tag{11}$$

$$\begin{aligned} x_{n,t}^{(E)} = W_E \cdot \mathrm{concat}\big(&X_{n,t,2,:},\ X_{n,t,4,:}, \\ &\hat{\omega}_a(n),\ \hat{\omega}_f(n,t)\big) + b_E \end{aligned} \tag{12}$$

$$\begin{aligned} x_{n,t}^{(H)} = W_H \cdot \mathrm{concat}\big(&X_{n,t,3,:},\ X_{n,t,5,:}, \\ &\hat{\omega}_a(n),\ \hat{\omega}_f(n,t)\big) + b_H \end{aligned} \tag{13}$$

$$x_{n,t}^{(C)} = W_C \cdot \mathrm{concat}\big(X_{n,t,6,:},\ \hat{\omega}_a(n),\ \hat{\omega}_f(n,t)\big) + b_C \tag{14}$$

where W_τ denotes the linear projection matrix independently maintained for each node type, and b_τ is the corresponding bias vector. The output dimension is uniformly set to d_{in}.

Each type of node receives only its corresponding data. Traffic nodes concatenate the speed and energy-consumption information of the three vehicle types. Electricity nodes concatenate the mobile electricity consumption of electric vehicles (Channel 2) and the load of fixed electrical equipment (Channel 4). Hydrogen nodes concatenate the mobile hydrogen consumption of hydrogen-powered vehicles (Channel 3) and the hydrogen flow rate of fixed hydrogen energy equipment (Channel 5). Heating/cooling nodes concatenate the load of fixed heating/cooling equipment (Channel 6).

As an illustration, the 'anatomical diagram' of the 17-dimensional features of the transportation node is shown in Figure 3. The node feature vector is structured as follows. The first 12 dimensions correspond to three vehicle channels, with four physical quantities per channel. The subsequent 4 dimensions constitute the grid facility profile, denoted as $\hat{\omega}_a$. The final dimension represents the grid busyness level, denoted as $\hat{\omega}_f$. For E and H nodes, the first 12 dimensions are replaced by their respective two channels ($2 \times 4 = 8$ dimensions), yielding a 13-dimensional vector when combined with the 4-dimensional facility profile and 1-dimensional busyness indicator ($8 + 4 + 1 = 13$). The C node, equipped with a single channel (4 dimensions), results in a 9-dimensional feature vector ($4 + 4 + 1 = 9$).

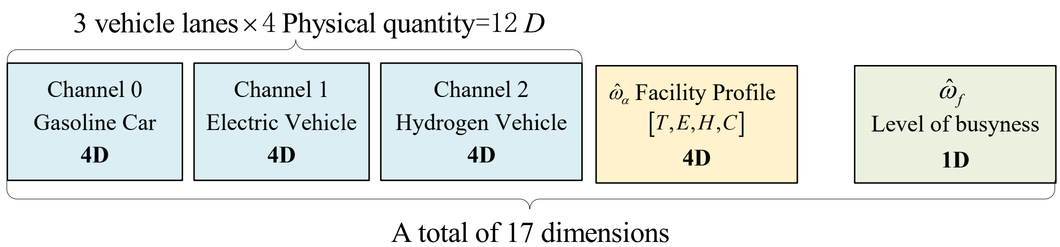

Figure 3. The 'anatomical diagram' of the 17-dimensional features of the transportation node.

In addition, all nodes are concatenated with the facility coverage vector $\hat{\omega}_a(n)$ of the corresponding grid, which serves as a static infrastructure profile, and the event intensity $\hat{\omega}_f(n,t)$, which provides a dynamic disturbance prior.

4. Hybrid HetGAT–Transformer Prediction Model

The multi-energy flow prediction task in transportation hubs involves two intertwined types of dependencies. The first is *spatial dependency*, where transportation, electricity, hydrogen, and thermal/cooling infrastructures form a complex heterogeneous topology in geographic space, and neighboring or coupled facilities exert mutual influences. The second is *temporal dependency*, where traffic and energy loads evolve over time and exhibit long-term trends, periodic patterns, and sudden disturbances. A single model, such as a pure graph neural network or a pure sequence model, is generally capable of capturing only one of these dependency types.

To address this issue, the proposed framework decouples the two dependencies into two serially connected encoders with orthogonal modeling objectives. The HetGAT encoder aggregates heterogeneous neighborhood information along the spatial dimension at each fixed time step [19], while the Transformer encoder captures historical temporal dependencies along the time dimension for each fixed node. The two modules are complementary rather than interchangeable. Their encoded representations are subsequently fused in a dual-task prediction head.

The model takes the standardized fourth-order tensor introduced in Section 3 as its sole input and generates end-to-end joint predictions of traffic states and multi-energy flow loads.

4.1. Overall Framework

The proposed model is organized as a pipeline of five stages, whose overall data flow is illustrated in Figure 4 and whose tensor shapes are listed in Table 2. In the first stage (Section 4.2), the fourth-order tensor is aggregated along the K dimension into per-grid features, mapped by type-specific projections into typed nodes, and assembled into a heterogeneous graph $\mathcal{G}$ according to the node and edge rules. The second stage (Section 4.3) applies two HetGAT layers independently at each time step and aggregates node features over the two-hop heterogeneous neighborhood to obtain the spatial embedding $Z_{n,t}^{(\tau)}$. In the third stage (Section 4.4), the T_{in} spatial embeddings of each node are stacked along the temporal dimension and passed through L Transformer layers, which capture long-range temporal dependencies and yield the node-level temporal embedding $H_n^{(\tau)}$. The fourth stage (Section 4.5) fuses the temporal embeddings of all typed nodes within the same grid and concatenates them with external auxiliary features to form the joint feature c_n. In the final stage (Section 4.6), c_n passes through a shared backbone that extracts cross-system coupling representations, from which a traffic head and an energy head produce their respective predictions.

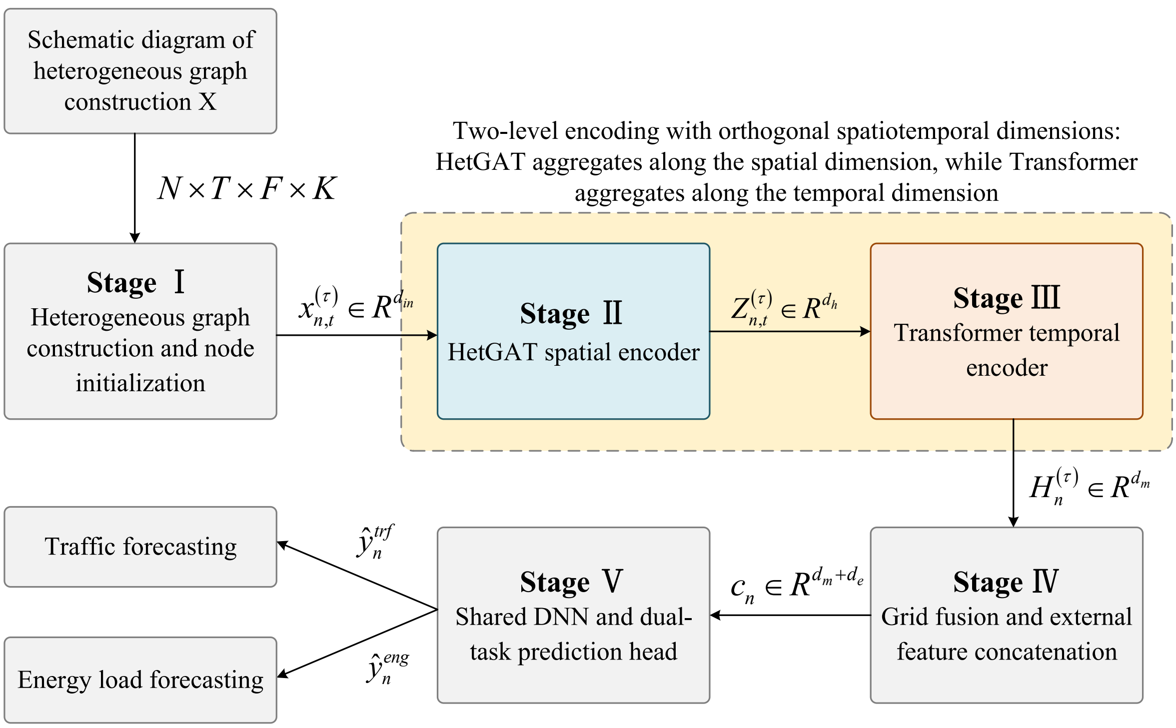

Figure 4. End-to-end data flow framework for HetGAT–Transformer hybrid prediction model.

Stage I performs the transformation from "tensor to graph", Stages II–III constitute the core spatiotemporal encoding modules of the model, and Stages IV–V complete the regression process from "graph to prediction". The following subsections are organized according to this sequence, and the tensor shape at each step can be cross-referenced row by row with Table 2.

Table 2. Tensor shapes and dimension symbols at each stage of the method.

Stage	Tensor Symbol	Shape
Raw 4D tensor	$\mathbf{X}$	$N \times T_{in} \times K \times F$
K-slot aggregation	$u_{n,t}$	F
Type-specific projection	$x_{n,t}^{(\tau)}$	d_{in}
HetGAT layer-1 output	$z_{n,t}^{(\tau,1)}$	d_h
HetGAT layer-2 output	$Z_{n,t}^{(\tau)}$	d_h
Temporal sequence stacking	$S_n^{(\tau)}$	$T_{in} \times d_h$
Transformer output	$H_n^{(\tau)}$	d_m
Grid-type fusion	g_n	d_m
Joint feature concatenation	c_n	$d_m + d_e$
DNN backbone expansion layer	$\eta_n^{(1)}$	d_{b_1}
DNN backbone output	η_n	d_b
Traffic prediction output	$\hat{y}_n^{trf}$	G_{trf}
Energy prediction output	$\hat{y}_n^{eng}$	G_{eng}
Number of node types	$M = \|\mathcal{T}_V\|$	4
Number of edge types	$R = \|\mathcal{T}_E\|$	5
Total number of graph nodes	$\|\mathcal{V}\|$	$\leq N \cdot M$

4.2. Stage I: Heterogeneous Graph Construction and Node Initialization

Stage I maps the standardized fourth-order tensor introduced in Section 3 into a heterogeneous graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and initializes node features, serving as the bridge between data representation and graph neural network modeling [20].

4.2.1. Node Definition

Each H3 grid derives at most $M = |\mathcal{T}_V| = 4$ typed nodes according to the types of facilities it contains, corresponding to traffic, electricity, hydrogen, and thermal/cooling facilities, respectively. If grid n does not contain facilities of a certain type τ, such as a water-only grid without hydrogen refueling stations, the corresponding typed node is not instantiated. The node set over the entire study area is denoted as:

$$\mathcal{V} = \bigcup_{n=1}^{N} \bigcup_{\tau \in \mathcal{T}_V^{(n)}} \{v_n^{(\tau)}\}, \quad |\mathcal{V}| \leq N \cdot M \tag{15}$$

where $\mathcal{T}_V^{(n)} \subseteq \mathcal{T}_V$ denotes the set of facility types actually contained in grid n. The upper bound $N \cdot M$ corresponds to the extreme case in which each grid contains all four types of facilities. In practice, most grids contain only one or two types of facilities, such as road-only grids that include only traffic nodes. Therefore, $|\mathcal{V}|$ is usually much smaller than $N \cdot M$.

The "four types of nodes" are directly determined by the physical composition of the multi-energy flow system in large-scale transportation hubs, corresponding to four non-overlapping subsystems:

(1) **Traffic subsystem (T)**: road networks, vehicle flows, signal control, etc., characterized by traffic volume, average speed, and lane occupancy. It constitutes the demand side of the hub.

(2) **Electricity subsystem (E)**: substations, charging piles, distribution lines, etc., characterized by active/reactive power load. It constitutes the energy-supply side for electrified equipment.

(3) **Hydrogen subsystem (H)**: hydrogen refueling stations, hydrogen storage tanks, hydrogen supply pipelines, etc., characterized by hydrogen flow rate and instantaneous consumption.In port scenarios dense with heavy-duty trucks, it is an independent carrier that cannot be merged into the electricity subsystem.

(4) **Thermal/cooling subsystem (C)**: cold storage, heating/cooling facilities, heat/cold storage tanks, etc., characterized by heating/cooling load.Port cold-chain warehousing and container refrigeration depend heavily on it.

Each node $v_n^{(\tau)}$ does not represent an individual physical entity. Instead, it serves as an aggregated proxy for the operational state of type-τ facilities within grid n at a given time step. This "grid $\times$ type" definition is naturally aligned with the H3 spatial indexing scheme introduced in Section 3. It abstracts heterogeneous physical entities, such as substations, hydrogen refueling stations, and thermal storage tanks, into unified typed graph-node interfaces, thereby enabling consistent processing by HetGAT.

4.2.2. Edge Definition

The edge-type set $\mathcal{T}_E$ contains $R = 5$ types:

Road edges (r_1) connect traffic nodes $v_n^{(T)}$ and $v_{n'}^{(T)}$ located in two different grids, if and only if the two grids are spatially adjacent in the H3 system and are connected through the road network.

Power-line edges (r_2) connect electricity nodes in two different grids, if and only if a physical power line exists between the corresponding grids.

Hydrogen-pipeline edges (r_3) connect hydrogen nodes in two different grids, if and only if a physical hydrogen supply pipeline exists between the corresponding grids.

Thermal/cooling network edges (r_4) connect thermal/cooling energy nodes in two different grids, if and only if a physical thermal/cooling network connection exists between the corresponding grids.

Cross-system coupling edges (r_5) connect two nodes of different types within the same H3 grid, such as $v_n^{(T)}$ and $v_n^{(E)}$, and explicitly represent the local physical conversion from traffic loads to energy consumption.

Edges r_1–r_4 are all "same-type, cross-grid" edges, which characterize the spatial propagation of the same type of energy or material flow, including traffic flow along roads, electricity flow along power lines, hydrogen flow along pipelines, and thermal/cooling energy flow along thermal/cooling networks. Therefore, the two endpoints of these edges must belong to different grids. In contrast, r_5 is a "cross-type, same-grid" edge that characterizes the local coupling through which transportation activities are converted into energy consumption. Since the supply side and demand side of this conversion process must be located at the same physical position, this type of edge is restricted to nodes within the same grid.

These two categories of edges are complementary: r_1–r_4 capture same-type propagation across grids, whereas r_5 captures cross-type conversion within the same grid. No "same-grid, same-type" edges are introduced. According to the aggregation strategy, each grid instantiates at most one node for each type, and the interactions among same-type entities within a grid have already been implicitly encoded in the aggregated node features.

Each edge (v_i, v_j) of type r is associated with a scalar prior weight $w_{ij}^{(r)} \in [0, 1]$, which is derived from the spatial element weights defined in Section 3.3:

$$w_{ij}^{(r)} = \begin{cases} \hat{\omega}_l^{(i,j)}, & r \in \{r_1, r_2, r_3, r_4\} \ \ \text{(line-shaped infrastructure)} \\[2ex] \dfrac{\hat{\omega}_a^{(n)} + \hat{\omega}_f^{(n)}}{2}, & r = r_5 \ \ \text{(nodes } v_i, v_j \text{ in the same grid } n\text{)} \end{cases} \tag{16}$$

For r_1–r_4, the weight is defined as the normalized length weight $\hat{\omega}_l$ of the linear infrastructure connecting the two grids. A longer road, pipeline, or network segment indicates stronger transmission capacity between the two endpoints. For r_5, the weight is defined as the arithmetic mean of the area weight $\hat{\omega}_a$ and the frequency weight $\hat{\omega}_f$ within the grid. Denser facilities and more frequent coupling events indicate more intensive local conversion. This design explicitly incorporates all spatial weights, including event frequency, into the heterogeneous graph. The weight $w_{ij}^{(r)}$ is further used as a multiplicative prior in attention normalization, as described in Section 4.3.

The one-hop neighborhood of node v is defined as the set of nodes that are directly connected to v through any type of edge in graph $\mathcal{G}$:

$$\mathcal{N}(v) = \left\{ v' \in \mathcal{V} \,\middle|\, \exists\, r \in \mathcal{T}_E,\ (v, v') \in \mathcal{E}_r \right\} \tag{17}$$

The sub-neighborhoods obtained by partitioning according to edge type r are denoted as $\mathcal{N}_r(v)$, satisfying $\mathcal{N}(v) = \bigcup_r \mathcal{N}_r(v)$. Taking the transportation node $v_n^{(T)}$ as an example, its one-hop neighborhood consists only of r_1 neighbors, namely transportation nodes in adjacent grids, and r_5 neighbors, namely electricity, hydrogen, heating, cooling nodes within the same grid. The sub-neighborhoods corresponding to all other edge types are empty. The schematic construction of the heterogeneous graph is shown in Figure 5.

Figure 5 contains 183 nodes and five types of edges, including 176 road edges, 3306 Electric edges, 3422 hydrogen-pipeline edges, 90 thermal edges, and 304 cross-system coupling edges.

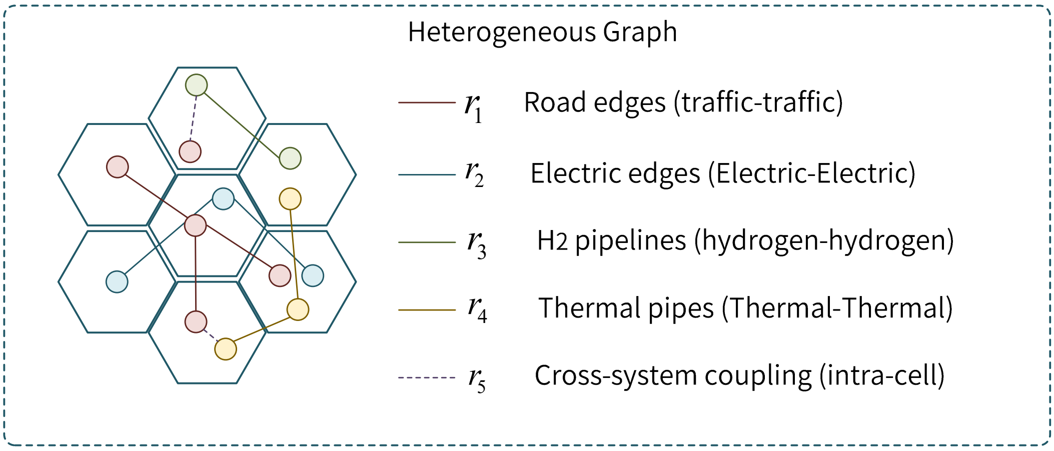

Figure 5. Schematic diagram of heterogeneous graph construction.

4.2.3. From Four-Dimensional Tensors to Initial Node Features

Starting from $u_{n,t} \in \mathbf{R}^F$, the initial features of each typed node are generated through the type-specific projection π_τ:

$$x_{n,t}^{(\tau)} = \pi_\tau(u_{n,t}) = W_\tau^{in}\, u_{n,t} + b_\tau^{in} \quad \in \ \mathbf{R}^{d_{in}} \tag{18}$$

where W_τ^{in} and b_τ^{in} denote the weight matrix and bias vector independently maintained for type τ, respectively. Equation (18) is the compact form of the type-specific construction detailed in Equations (11)–(14), in which the channels relevant to each node type, together with the area and event-intensity weights of grid n, are projected by a type-specific matrix to the common dimension d_{in}. π_τ routes the raw physical quantities according to their type, so that π_T primarily preserves the components related to speed and traffic flow, while π_E preserves the electricity consumption component and π_H and π_C are defined analogously. $x_{n,t}^{(\tau)}$ serves as the input to Stage II, and each dimensional transformation involved can be traced in Table 2.

4.3. Stage II: HetGAT Spatial Encoder

Stage II takes the typed node features $\{x_{n,t}^{(\tau)}\}$ produced by Stage I as input and operates independently at each time step, generating for each node a spatial embedding $Z_{n,t}^{(\tau)} \in \mathbf{R}^{d_h}$ that integrates both its own state and the heterogeneous neighborhood state. By stacking two layers, the receptive field is expanded from one-hop to two-hop neighborhoods. This encoding only characterizes "which nodes influence which other nodes" on the graph at the same time instant, without involving temporal dependencies. Figure 6 illustrates the structure of one layer of HetGAT. A two-layer HetGAT is constructed by stacking two identical instances of the aforementioned architecture in a sequential manner, where the output node representations $\mathbf{z}_v$ from the first layer serve as the input features $\mathbf{x}_v$ for the second layer. This stacking effectively extends the receptive field from one-hop to two-hop neighborhoods.

The necessity of adopting heterogeneous, rather than homogeneous, graph attention lies in two aspects. First, the feature semantics of different node types are not directly comparable. For instance, the numerical value "speed $= 50$" of a transportation node and "power $= 50\,\mathrm{kW}$" of an electricity node may be identical, yet their physical meanings are fundamentally different. Therefore, each node type must maintain an independent projection matrix W_τ. Second, different edge types convey distinct semantics, since road edges r_1 represent physical connectivity, whereas cross-system edges r_5 represent energy conversion relationships. Hence, each edge type must maintain independent attention parameters $(\mathbf{a}_r, W_{ar}, W_{br})$. Node-specific projection and edge-type-specific attention together constitute the "dual heterogeneity" mechanism of HetGAT. The two-layer structures are formally identical, with their receptive fields corresponding to one-hop and two-hop neighborhoods, respectively.

It should be particularly noted that the proposed HetGAT adopts *single-head* additive attention for each edge type, as defined in Equations (20)–(23). Its heterogeneity is reflected in the independent maintenance of attention parameters for different edge types, rather than in multi-head parallelization. This differs from the multi-head self-attention employed by the Stage III Transformer (Section 4.4). Given the existing R sets of edge-type-specific parameters, the single-head design is sufficient to capture the differentiated importance of heterogeneous neighborhoods while reducing both parameter size and computational cost. The extension to multi-head HetGAT is left for future work.

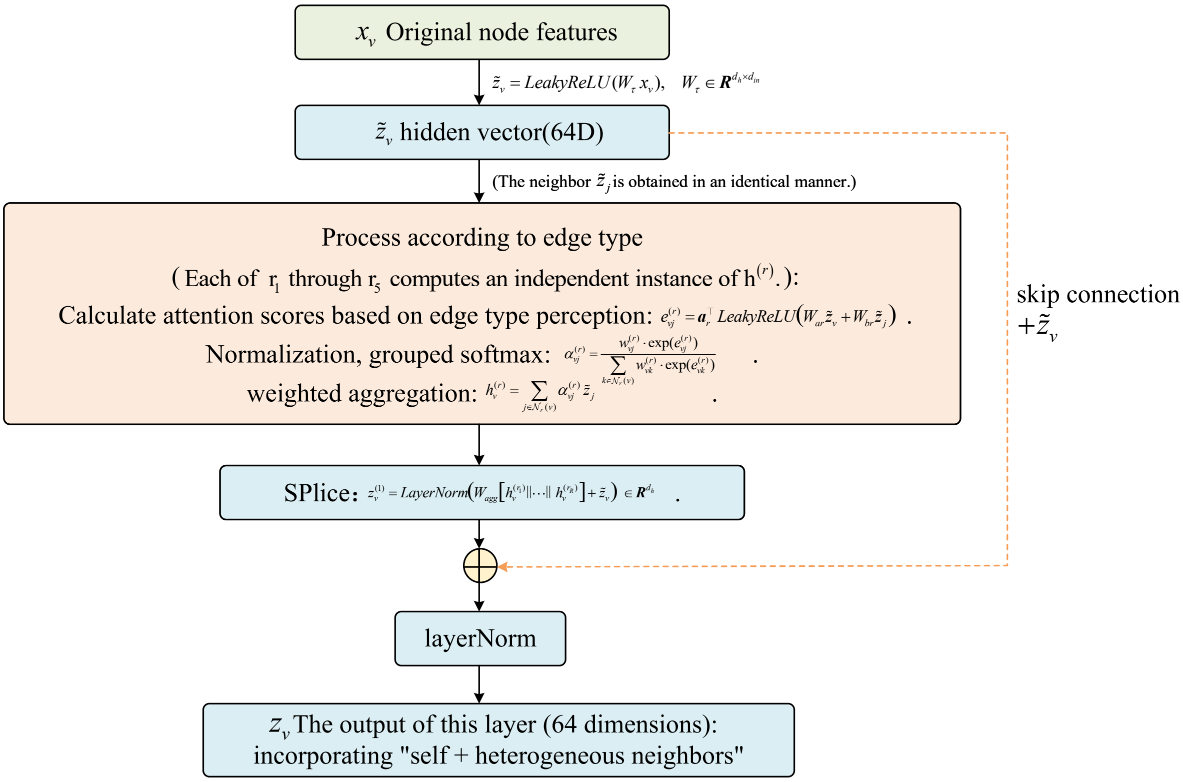

Figure 6. The structure of one layer of HetGAT.

4.3.1. Layer 1: One-Hop Neighborhood Aggregation

For a node v of type τ, a type-specific nonlinear projection is first performed to compress the d_{in}-dimensional input into the common hidden dimension d_h:

$$\tilde{z}_v = LeakyReLU(W_\tau x_v), \quad W_\tau \in \mathbf{R}^{d_h \times d_{in}} \tag{19}$$

where W_τ denotes the projection matrix specific to type τ, and $LeakyReLU(x) = \max(0.01x, x)$ is an element-wise nonlinear activation function. Subsequently, for each edge type $r \in \mathcal{T}_E$ and each neighbor $j \in \mathcal{N}_r(v)$, an edge-type-aware attention score is computed as follows:

$$e_{vj}^{(r)} = \mathbf{a}_r^\top \, LeakyReLU\big(W_{ar}\tilde{z}_v + W_{br}\tilde{z}_j\big) \tag{20}$$

where $W_{ar}, W_{br} \in \mathbf{R}^{d_h \times d_h}$ denote the projection matrices associated with the source node v and the target node j, respectively, explicitly encoding the role asymmetry between the two endpoints of a directed edge. $\mathbf{a}_r \in \mathbf{R}^{d_h}$ is a learnable attention vector specific to edge type r, which compresses the transformed representation into a scalar attention score $e_{vj}^{(r)}$ through an inner product operation. This score characterizes the unnormalized importance of neighbor j to node v under the semantic context of edge type r. Since each edge type maintains an independent parameter set $(\mathbf{a}_r, W_{ar}, W_{br})$, the criterion for determining an "important neighbor" may differ between r_1 and r_5.

During the normalization stage, the edge prior weight $w_{vj}^{(r)}$ (defined in Equation (16)) is incorporated into the Softmax function as a multiplicative bias:

$$\alpha_{vj}^{(r)} = \frac{w_{vj}^{(r)} \cdot \exp(e_{vj}^{(r)})}{\sum_{k\in\mathcal{N}_r(v)} w_{vk}^{(r)} \cdot \exp(e_{vk}^{(r)})} \tag{21}$$

$\alpha_{vj}^{(r)} \in (0,1)$ and is normalized within edge type r. The multiplicative prior $w_{vj}^{(r)}$ enables the spatial factor information defined in Section 3.3, including event frequency, to be injected into the attention distribution in a differentiable manner. As a constant, it does not participate in training and therefore does not compromise the learnability of the attention mechanism. Subsequently, weighted aggregation is performed separately for each edge type:

$$h_v^{(r)} = \sum_{j\in\mathcal{N}_r(v)} \alpha_{vj}^{(r)} \, \tilde{z}_j \tag{22}$$

A d_h-dimensional aggregation vector $h_v^{(r)}$ is obtained for each of the R edge types. If $\mathcal{N}_r(v) = \emptyset$, then $h_v^{(r)} = \mathbf{0}$. Finally, the R aggregation results are concatenated and compressed back to d_h dimensions through the linear dimensionality-reduction matrix $W_{agg} \in \mathbf{R}^{d_h \times (R \cdot d_h)}$. The resulting representation is then combined with the node's own feature through a residual connection and layer normalization [21]:

$$z_v^{(1)} = LayerNorm\Big(W_{agg}\big[h_v^{(r_1)} \| \cdots \| h_v^{(r_R)}\big] + \tilde{z}_v\Big) \in \mathbf{R}^{d_h} \tag{23}$$

The residual connection [22] preserves the node's self-information and stabilizes training. $z_v^{(1)}$ encodes, in a distributed representation, both the state of node v itself and the weighted states of all types of neighbors within its one-hop neighborhood.

4.3.2. Layer 2: Two-Hop Neighborhood Aggregation

The second layer has exactly the same structure as the first layer, and likewise performs Equations (19)–(23). The only difference is that the input is replaced by the first-layer outputs $z_v^{(1)}$ instead of the initial node features x_v, and the output is denoted as:

$$Z_v \equiv z_v^{(2)} \in \mathbf{R}^{d_h} \tag{24}$$

The second-layer parameters $(W'_\tau, \mathbf{a}'_r, W'_{ar}, W'_{br}, W'_{agg})$ are maintained independently and are not shared with those of the first layer. Since $z_j^{(1)}$ carried by each neighbor j already contains information from the one-hop neighborhood of j itself, the second aggregation enables v to indirectly acquire information from all nodes within its two-hop neighborhood:

$$\mathcal{N}^{(2)}(v) = \bigcup_{j \in \mathcal{N}(v)} \mathcal{N}(j) \supseteq \mathcal{N}(v) \tag{25}$$

The two-layer architecture adopted in this study represents a trade-off between receptive field expansion and over-smoothing. Two layers are sufficient to cover typical multi-energy flow coupling paths in large-scale hubs, such as "road $\rightarrow$ substation within the same grid $\rightarrow$ hydrogen station in an adjacent grid". In contrast, excessive layers may cause node embeddings to become overly similar, thereby reducing their discriminative capacity. The HetGAT outputs independently obtained over the T_{in} time steps are organized by node and time to form the spatial embedding sequence $\{Z_{n,t}^{(\tau)}\}_{t=1}^{T_{in}}$, which serves as the input to Stage III.

4.4. Stage III: Transformer Temporal Encoder

Stage III takes as input the spatial embedding sequence $\{Z_{n,t}^{(\tau)}\}_{t=1}^{T_{in}}$ produced by Stage II over the T_{in} time steps. If HetGAT provides a hub-wide spatial snapshot at each time instant, the Transformer models the T_{in} snapshots sequentially along the temporal dimension. Through the self-attention mechanism, each node can adaptively determine which historical time steps should be emphasized for prediction, for example, assigning larger weights to historical periods at the same time of day and to the corresponding period in the previous week when forecasting the load of a certain period on the following day, while assigning smaller weights to random disturbances. Compared with Long Short-Term Memory (LSTM), which traverses the sequence step by step in chronological order, the Transformer can model the entire historical segment simultaneously and freely select the most informative time steps, making it more suitable for capturing long-range temporal dependencies.

The Transformer operates independently along the temporal dimension for each node. Together with HetGAT, it forms a two-stage spatiotemporal encoding scheme with orthogonal data flows. HetGAT aggregates spatial dependencies among nodes at a fixed time instant, whereas the Transformer aggregates temporal dependencies among time steps for a fixed node. The two modules focus on mutually orthogonal directions and are not substitutes for each other. Consequently, the final node-level temporal embedding $H_n^{(\tau)}$ simultaneously distills both spatial patterns and temporal trends.

4.4.1. Node Temporal Sequence and Positional Encoding

For each node (n, τ), its HetGAT outputs are stacked along the T_{in} time steps to form a temporal matrix:

$$S_n^{(\tau)} = \big[Z_{n,1}^{(\tau)}, Z_{n,2}^{(\tau)}, \ldots, Z_{n,T_{in}}^{(\tau)}\big]^\top \in \mathbf{R}^{T_{in} \times d_h} \tag{26}$$

Each row corresponds to a spatial snapshot at a historical time step. If the Transformer model dimension d_m is inconsistent with d_h, a linear projection $W_{in}^T \in \mathbf{R}^{d_m \times d_h}$ is first applied to map the representation into the d_m-dimensional space, followed by the addition of sinusoidal/cosine positional encoding:

$$PE(k,2i) = \sin\Big(\frac{k}{10000^{2i/d_m}}\Big), \quad PE(k,2i+1) = \cos\Big(\frac{k}{10000^{2i/d_m}}\Big) \tag{27}$$

where $k = 1, \ldots, T_{in}$ denotes the time-step index, and $i = 0, \ldots, d_m/2 - 1$ denotes the dimension index. The positional encoding is directly added to the input matrix ($S \leftarrow S + PE$), enabling global self-attention to implicitly perceive temporal order. Otherwise, the Transformer would treat the input as an unordered set.

4.4.2. Multi-Head Self-Attention

The core of the encoder is scaled dot-product attention and its multi-head extension:

$$Attn(Q,K,V) = softmax\Big(\frac{QK^\top}{\sqrt{d_k}}\Big)V, \tag{28}$$

$$MH(Q,K,V) = Concat(head_1, \ldots, head_H)\, W^O, \tag{29}$$

$$head_i = Attn(QW_i^Q, KW_i^K, VW_i^V). \tag{30}$$

$QK^\top$ computes the pairwise similarity between all time steps, and the result is scaled by $\sqrt{d_k}$ to prevent Softmax saturation. $W_i^Q, W_i^K, W_i^V \in \mathbf{R}^{d_m \times d_k}$ denote the projection matrices of the i-th head, where $d_k = d_m/H$ is the dimensionality of a single head. The H heads are computed independently, concatenated, and then linearly mixed through $W^O \in \mathbf{R}^{d_m \times d_m}$, enabling the model to attend simultaneously to different temporal patterns, such as short-term fluctuations and periodic trends. In self-attention, Q, K, V are all obtained from the same input sequence through three sets of learnable linear transformations. In this study, L encoder layers are stacked, each consisting of a multi-head self-attention sublayer and a feed-forward network (FFN) sublayer, with residual connections and layer normalization employed to stabilize training.

4.4.3. Temporal Pooling Output

After the L encoder layers, mean pooling is applied to the outputs of all time steps to obtain a fixed-dimensional node-level temporal embedding.

$$H_n^{(\tau)} = \frac{1}{T_{in}} \sum_{t=1}^{T_{in}} \big[Encoder^L(S_n^{(\tau)})\big]_{t,:} \;\in\; \mathbf{R}^{d_m} \tag{31}$$

Here, $[\cdot]_{t,:}$ denotes the t-th row of the encoder output matrix. $H_n^{(\tau)}$ integrates the evolution of spatiotemporal features of node (n, τ) over the past T_{in} time steps, serving as the fundamental unit for Stage IV fusion and final prediction.The reason for using mean pooling here is that it can aggregate information from the entire historical window without introducing additional trainable parameters, and is less sensitive to fluctuations in individual terminal time steps compared to the last step of readout. Compared to learning time aggregation, it also provides parameter efficient representation and helps to limit additional model complexity under available training data.

Algorithm 1 summarizes Stages I–III in pseudocode, covering the complete forward process from the original four-dimensional tensor, through type-based routing and HetGAT spatial encoding, to Transformer-based temporal encoding, and finally to the output of node-level temporal embeddings.

4.5. Stage IV: Grid-Level Fusion and External Feature Concatenation

Stage IV fuses the temporal embeddings $H_n^{(\tau)}$ of up to M typed nodes within the same grid, obtained from Stage III, into a unified grid-level representation. In this study, a learnable type-weighted fusion strategy is adopted:

$$g_n = \sum_{\tau \in \mathcal{T}_V^{(n)}} \beta_\tau\, H_n^{(\tau)} \;\in\; \mathbf{R}^{d_m} \tag{32}$$

Here, $\beta_\tau \geq 0$ denotes the *globally shared* learnable type weight, which is normalized once over the $M = 4$ node types via Softmax ($\sum_{\tau=1}^{M} \beta_\tau = 1$). All grids share the same set of weights $\{\beta_\tau\}$, enabling the model to adaptively learn the relative importance of traffic, power, hydrogen, and heating/cooling facilities to the overall grid state. For grids containing only a subset of node types, the summation is performed only over the actually existing types $\mathcal{T}_V^{(n)}$ within that grid. The weighted summation does not change the feature dimensionality, which remains d_m. The use of a single globally shared type-weight vector is a simplifying assumption that reduces model complexity and promotes stable parameter learning across grids. Nevertheless, the relative importance of different node types may vary across spatial locations and operating conditions; therefore, grid-dependent or context-aware adaptive weighting will be investigated in future work.Subsequently, g_n is concatenated with the external auxiliary feature $e_t^{ext} \in \mathbf{R}^{d_e}$:

$$c_n = \left[g_n \,\|\, e_t^{ext}\right] \quad \in \quad \mathbf{R}^{d_m+d_e} \tag{33}$$

Here, e_t^{ext} denotes the global external feature at time t, including hour/week periodic encodings, holiday indicators, and meteorological variables. All grids share the same vector at a given time step. If a certain feature, such as the vehicle-type proportion, is aggregated at the grid level, the corresponding component is indexed by the grid. c_n is then fed into Stage V as the joint feature vector. Figure 7 illustrates the entire process from nodes to predictions.

Algorithm 1: Forward Inference—Stages I–III: Feature Extraction and Spatio-Temporal Encoding

Input: Four-dimensional tensor $\mathbf{X} \in \mathbf{R}^{N\times T_{in}\times K\times F}$; heterogeneous graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with edge weights $\{w_{ij}^{(r)}\}$; spatial weights $\{W_n\}$
Output: Node-level temporal embeddings $\{H_n^{(\tau)}\}$

`/* Stage I: Tensor → Typed Node Features */`
for $n = 1, \ldots, N$; $t = 1, \ldots, T_{in}$ **do**
  $u_{n,t} \leftarrow$ slot-weighted aggregation; `// Equation (6)`
  $u_{n,t} \leftarrow W_n \cdot u_{n,t}$; `// spatial prior scaling`
  foreach *node type* $\tau \in \mathcal{T}_V$ **do**
    $x_{n,t}^{(\tau)} \leftarrow \pi_\tau(u_{n,t})$; `// Equation (18)`

`/* Stage II: HetGAT Spatial Encoding (independent per time step) */`
for $t = 1, \ldots, T_{in}$ **do**
  foreach *node* $v \in \mathcal{V}$ **do**
    $\tilde{z}_v \leftarrow LeakyReLU(W_\tau\, x_v)$ **foreach** *edge type* $r \in \mathcal{T}_E$ **do**
      Compute attention score $e_{vj}^{(r)}$; `// Equation (20)`
      Edge-weighted Softmax normalization; `// Equation (21)`
      $h_v^{(r)} \leftarrow \sum_{j\in\mathcal{N}_r(v)} \alpha_{vj}^{(r)} \tilde{z}_j$
    $z_v^{(1)} \leftarrow LayerNorm(W_{agg}[h_v^{(r_1)} \| \cdots \| h_v^{(r_R)}] + \tilde{z}_v)$
  Repeat the above steps with $\{z_v^{(1)}\}$ as input $\rightarrow Z_{n,t}^{(\tau)}$; `// second layer`

`/* Stage III: Transformer Temporal Encoding (independent per node) */`
foreach *node* (n, τ) **do**
  $S_n^{(\tau)} \leftarrow [Z_{n,1}^{(\tau)}, \ldots, Z_{n,T_{in}}^{(\tau)}] + PE$ **for** $\ell = 1, \ldots, L$ **do**
    $S_n^{(\tau)} \leftarrow TransformerLayer(S_n^{(\tau)})$
  $H_n^{(\tau)} \leftarrow MeanPool_t(S_n^{(\tau)})$
return $\{H_n^{(\tau)}\}$;

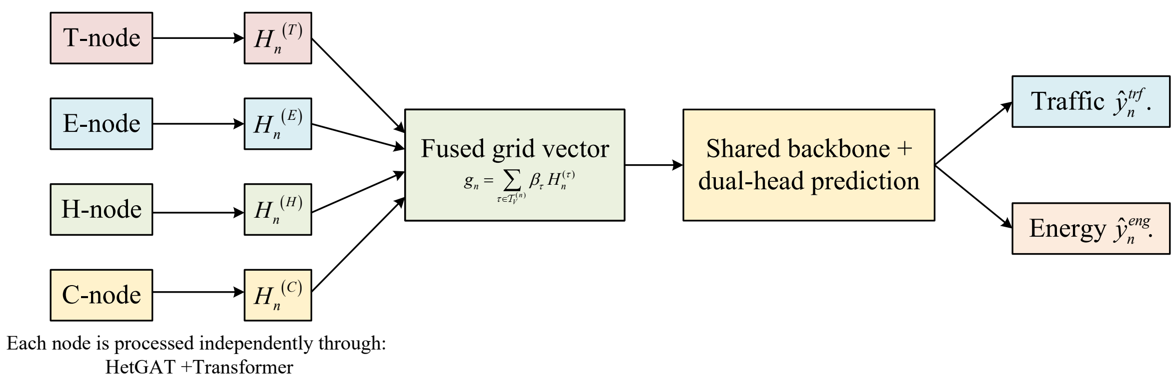

Figure 7. The entire process from nodes to predictions.

4.6. *Stage V: Shared DNN Backbone and Dual-Task Prediction Heads*

Stage V takes the joint feature c_n as input. A shared backbone is first employed to extract cross-system coupled representations, followed by two task-specific heads for regression outputs.

4.6.1. Shared DNN Backbone

c_n is passed through a two-layer shared Deep Neural Network (DNN) backbone to progressively extract cross-system coupled representations:

$$\eta_n^{(1)} = ReLU(W_1\, c_n + b_1) \;\; \in \; \mathbf{R}^{d_{b_1}} \tag{34}$$

$$\eta_n = ReLU(W_2\, \eta_n^{(1)} + b_2) \;\; \in \; \mathbf{R}^{d_b} \tag{35}$$

Here, $W_1 \in \mathbf{R}^{d_{b_1} \times (d_m + d_e)}$ projects the joint feature into a d_{b_1}-dimensional space, while $W_2 \in \mathbf{R}^{d_b \times d_{b_1}}$ further compresses it to d_b dimensions, with $d_{b_1} > d_b$. This bottleneck structure of "expansion followed by compression" enables the model to explore high-order nonlinear coupling relationships in a larger feature space during the expansion stage, while forcing it to retain the most discriminative representations and discard redundant information during the compression stage.

4.6.2. Dual-Task Prediction Heads

At the end of the model, the network branches into two independent linear prediction heads.

$$\hat{y}_n^{trf} = W_{trf}\, \eta_n + b_{trf} \;\; \in \; \mathbf{R}^{G_{trf}} \tag{36}$$

$$\hat{y}_n^{eng} = W_{eng}\, \eta_n + b_{eng} \;\; \in \; \mathbf{R}^{G_{eng}} \tag{37}$$

The traffic task head outputs a G_{trf}-dimensional prediction of traffic states, including traffic flow, average speed, lane occupancy, and other variables over the future T_{out} steps. The energy task head outputs a G_{eng}-dimensional prediction of multi-energy flow loads, including power load, hydrogen consumption, heating/cooling loads, and related variables. The two task heads share the parameters of the preceding backbone layers, forcing the backbone network to learn cross-system coupled representations that are beneficial to both tasks during backpropagation, thereby explicitly establishing a bidirectional interaction pathway between traffic flow and energy load.

To stabilize training and incorporate a persistence prior, this study does not directly regress absolute values. Instead, the task heads output *residual increments* relative to the observation at the last historical time step t_0. Taking the traffic task as an example, let the output of Equation (36) be the increment $\Delta\hat{y}_n^{trf}$. The final prediction is then given by:

$$\hat{y}_n^{trf} = y_{n,t_0}^{trf} + \Delta\hat{y}_n^{trf}, \qquad \hat{y}_n^{eng} = y_{n,t_0}^{eng} + \Delta\hat{y}_n^{eng} \tag{38}$$

Here, y_{n,t_0} denotes the true observation at the last time step of the input window, which serves as the persistence baseline and is broadcast along T_{out}. This residual parameterization is equivalent to imposing a strong prior that "the future is approximately equal to the present", such that the model only needs to learn the deviation from persistence, thereby substantially reducing the optimization difficulty. When the increment term is identically zero, the prediction degenerates into the persistence baseline.

4.6.3. Uncertainty-Weighted Joint Loss Function

In dual-task learning, a fixed-weight loss function may cause one task to dominate the gradient updates. Therefore, this study adopts an automatic weighting strategy based on homoscedastic uncertainty, allowing the model to adaptively learn the relative importance of the two tasks [23]. First, the subtask losses for traffic and energy prediction are defined separately, both as mean squared errors computed only over the set of *active grids* $\mathcal{A}$ containing nodes, where $|\mathcal{A}| = N_{active}$. The traffic task loss is defined as:

$$\mathcal{L}_{trf} = \frac{1}{N_{active} T_{out}} \sum_{n \in \mathcal{A}} \sum_{t=1}^{T_{out}} \left\| \hat{y}_{n,t}^{trf} - y_{n,t}^{trf} \right\|^2 \tag{39}$$

The energy task loss is defined as:

$$\mathcal{L}_{eng} = \frac{1}{N_{active} T_{out}} \sum_{n \in \mathcal{A}} \sum_{t=1}^{T_{out}} \left\| \hat{y}_{n,t}^{eng} - y_{n,t}^{eng} \right\|^2 \tag{40}$$

Here, $\hat{y}_{n,t}^{trf}$ and $\hat{y}_{n,t}^{eng}$ denote the predicted values of the traffic and energy tasks, respectively, for grid n at the future prediction step t, while $y_{n,t}^{trf}$ and $y_{n,t}^{eng}$ denote the corresponding ground-truth values. The summation is performed over the active grid set $\mathcal{A}$ and the future T_{out} prediction steps. Empty grids without any nodes are excluded from loss computation to prevent a large number of empty grids from diluting the gradient signals.

On the basis of the two subtask losses, the uncertainty-weighted joint loss is defined as:

$$\mathcal{L} = \frac{1}{2\sigma_1^2}\mathcal{L}_{trf} + \frac{1}{2\sigma_2^2}\mathcal{L}_{eng} + \ln\sigma_1 + \ln\sigma_2 + \lambda\|\Theta\|_2^2 \tag{41}$$

Each term can be interpreted as follows: $\frac{1}{2\sigma_1^2}\mathcal{L}_{trf}$ and $\frac{1}{2\sigma_2^2}\mathcal{L}_{eng}$ are the weighted losses for the traffic and energy tasks, respectively. A larger value of σ corresponds to a smaller task weight, thereby reducing the interference of noisy gradients from one task on the optimization of the other. The terms $\ln\sigma_1 + \ln\sigma_2$ act as regularizers that prevent σ from increasing indefinitely and trivially driving the loss toward zero. $\lambda\|\Theta\|_2^2$ denotes the L_2 weight decay term, where Θ represents the set of all model parameters and λ is the regularization coefficient. In implementation, the uncertainty is parameterized using the log-variance $s = \ln\sigma^2$ (i.e., $\frac{1}{2\sigma^2} = \frac{1}{2}e^{-s}$ and $\ln\sigma = \frac{1}{2}s$), and s is clipped to the range $[-4, 2]$ to ensure numerical stability. The parameters σ_1 and σ_2 are learnable task-uncertainty variables that are optimized jointly with all other model parameters.

Algorithm 2 summarizes Stages IV–V in pseudocode, covering the complete process from node-level temporal embeddings through grid-level fusion, external feature concatenation, the shared DNN backbone, and finally the dual-task outputs. During training, the uncertainty-weighted joint loss defined in Equation (41) is applied at the end of the algorithm, followed by backpropagation. Together, Algorithms 1 and 2 constitute the complete forward inference pipeline of the proposed model, and the tensor shape corresponding to each operation can be found in Table 2.

Algorithm 2: Forward Inference—Stages IV–V: Grid Fusion and Dual-Task Prediction

Input: Node-level temporal embeddings $\{H_n^{(\tau)}\}$ (output of Algorithm 1); external features $\{e_n^{ext}\}$
Output: Traffic predictions $\{\hat{y}_n^{trf}\}_{n=1}^N$, energy predictions $\{\hat{y}_n^{eng}\}_{n=1}^N$

`/* Stage IV: Grid Fusion + External Features */`
for $n = 1, \ldots, N$ **do**
  $g_n \leftarrow \sum_{\tau \in \mathcal{T}_V^{(n)}} \beta_\tau H_n^{(\tau)}$; `// type-weighted fusion`
  $c_n \leftarrow [g_n \,\|\, e_n^{ext}]$; `// concatenate external features`

`/* Stage V: Shared DNN + Dual-Task Heads */`
for $n = 1, \ldots, N$ **do**
  $\eta_n \leftarrow ReLU(W_2\, ReLU(W_1\, c_n + b_1) + b_2)$; $\hat{y}_n^{trf} \leftarrow W_{trf}\,\eta_n + b_{trf}$; `// traffic prediction head`
  $\hat{y}_n^{eng} \leftarrow W_{eng}\,\eta_n + b_{eng}$; `// energy prediction head`
return $\{\hat{y}_n^{trf}\}, \{\hat{y}_n^{eng}\}$;

5. Experiments and Analysis

5.1. Experimental Setup

Dataset and Scenario. The experimental data come from the SUMO simulation scenario constructed in Section 2, covering a large integrated transportation hub and its surrounding multi-energy flow infrastructure. The original data are sampled at a granularity of 5 minutes (288 steps per day). After spatiotemporal alignment, cleaning, H3 grid indexing, and fourth-order tensor representation in Sections 2 and 3, a standardized four-dimensional tensor $\mathbf{X} \in \mathbf{R}^{N \times T_{in} \times K \times F}$ is obtained.The experimental scenario spans 30 days with a total of $T = 8640$ time periods at a temporal resolution of 5 minutes (288 steps per day). The underlying road network is encoded as a heterogeneous graph consisting of 183 nodes distributed across four node types, whose input feature dimensions are 17, 13, 13, and 9, respectively, and five edge types with cardinalities of 176, 3,306, 3,422, 90, and 304 edges, respectively. Spatial regions are discretized using H3 hierarchical grids, yielding N_{active} = 80 active grid cells that participate in traffic state prediction. The full dataset is partitioned using a whole-day holdout strategy into a training set of 28 days (8,052 samples), a validation set of 1 day (288 samples), and a test set of 1 day (277 samples).

Task Definition. Taking a historical input window of $T_{in} = 12$ steps (i.e., 1 hour) as input, the model predicts the future $T_{out} = 12$ steps (i.e., 1 hour). The traffic task regresses $G_{trf} = 1$ output dimensions (flow), and the energy task regresses $G_{eng} = 3$ output dimensions (electricity, hydrogen, and thermal/cooling load). The model jointly predicts the following four physical quantities:

- **Traffic side**: traffic flow (unit: veh / 5 min, referred to as Traffic Flow);
- **Energy side**: electricity load (unit: kWh), hydrogen consumption (unit: kg), and thermal/cooling load (unit: kWh).

Both task types are jointly output by the shared backbone network within a single forward pass, and the predictions are evaluated after being de-normalized to their respective physical units.

Data Partitioning and Evaluation. The training, validation, and test sets are partitioned chronologically in a 28:1:1 ratio, and the test window is kept strictly later than the training window to avoid information leakage. Given the limited sample size of a single scenario, two evaluation protocols are adopted. The first is a holdout protocol that uses a fixed temporal split for rapid evaluation. The second is a walk-forward protocol that performs K-fold (default $K = 4$) forward validation with a rolling starting point and reports the mean and standard deviation for more robust results. The same set of metrics is reported under both protocols.

Evaluation Metrics. In physical units after denormalization, the Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and Coefficient of Determination (R^2) are reported separately for the traffic and energy tasks. MAE measures the average absolute deviation between predicted and true values, reflecting typical prediction accuracy. RMSE penalizes larger errors more heavily, capturing the stability of predictions under peak or anomalous conditions. R^2 indicates the proportion of variance in the target variable explained by the model, providing an intuitive measure of overall fitting quality.

Implementation Details. The model dimensions are set as $d_{in} = 32$, $d_h = d_m = 64$. The HetGAT comprises 2 layers, and the Transformer consists of $L = 2$ layers with $H = 4$ attention heads and an FFN dimension of 128. The backbone expansion and compression dimensions are $d_{b_1} = 128$ and $d_b = 64$, respectively. The external feature dimension is $d_e = 4$, corresponding to sinusoidal/cosine encodings of time-of-day and day-of-week. Dropout is applied at a rate of 0.3. The model is optimized using Adam with an initial learning rate of 10^{-3}, which decays to 10^{-6} via cosine annealing. Weight decay is set to $\lambda = 10^{-4}$, the batch size is 16, and gradient clipping is applied with a threshold of 1.0. Training proceeds for a maximum of 200 epochs with early stopping on the validation set (patience = 30 epochs). The random seed is fixed at 42. All experiments are conducted on a single NVIDIA GeForce RTX 4060 GPU with 8 GB of dedicated memory. The implementation is built upon PyTorch 2.10.0 with CUDA 13.0, running on Python 3.14.1 (64-bit, Windows).

5.2. Ablation Experiment

To clarify the independent contributions of each module, we start from the complete model and replace/remove key components one by one (keeping other settings unchanged). The ablation dimensions and corresponding hypotheses are as follows:

(a) Homogeneous GAT: Remove the distinction parameters between node types and edge types, and degenerate HetGAT into a homogeneous graph attention network to verify the necessity of the dual-heterogeneous mechanism.
(b) LSTM: Retain the heterograph structure but replace the Transformer temporal encoder with an LSTM, to verify the advantage of the global self-attention mechanism over the stepwise recursive mechanism in long-range temporal dependency modeling.
(c) Single-task: Split the transportation and energy tasks into two independent models for separate training, eliminating shared backbones, to verify the positive transfer effect of multi-task joint learning.

The results are summarized in Table 3, with each row representing an ablation variant. Relative changes in the three task metrics are presented against the full model to quantify the contribution of each module. The prediction curves are shown in Figures 8–11.

Table 3. Ablation experiment comparison (traffic and energy tasks reported separately; R^2: higher is better; RMSE & MAE: lower is better).

Method	**Traffic Flow**			**Electricity Load**			**Hydrogen Load**			**Thermal-Cooling Consumption**		
	R^2	**RMSE**	**MAE**	R^2	**RMSE**	**MAE**	R^2	**RMSE**	**MAE**	R^2	**RMSE**	**MAE**
HomoGAT	0.9738	0.2701	0.2089	0.7830	1.7450	1.1607	0.9647	0.0235	0.0181	0.9593	1.3850	0.9418
LSTM	0.9750	0.2642	0.2010	0.7909	1.7130	1.1280	0.9528	0.0271	0.0214	0.9533	1.4830	0.9800
Single-task	0.9738	0.2701	0.2072	0.7166	1.9940	1.4420	0.9333	0.0323	0.0243	0.9519	1.5050	1.0130
Our model	0.9763	0.2468	0.1921	0.8088	1.6380	1.0760	0.9644	0.0236	0.0183	0.9583	1.4010	0.9448

Figure 8 illustrates the prediction curves of the complete model (HetGAT + Transformer + multi-task joint prediction) presented in this paper for the four tasks at the first prediction step on the test set, serving as a unified benchmark for subsequent ablation studies and baseline comparisons. Overall, the predicted values (represented by the red line) closely align with the actual values (represented by the black line) in terms of both global trends and local fluctuations. In terms of traffic flow, the prediction curve closely follows the upward and downward trends of the actual values, with only minor lags observed at a few peak values. The root mean squared error (RMSE) is

0.2468 and the coefficient of determination (R^2) is 0.9763, indicating an excellent fitting performance. Electricity load is significantly influenced by random disturbances and exhibits stronger high-frequency fluctuations than the other energy variables. The proposed model achieves an RMSE of 1.6380 and an R^2 of 0.8088 for electricity-load prediction. Its relatively lower R^2 can be attributed to the superposition of mobile electricity demand from electric vehicles and fixed-equipment loads, which introduces stronger short-term variability and heterogeneous operating patterns. Although the model accurately captures the overall trend, some smoothing remains in high-frequency fluctuation segments. Hydrogen consumption demonstrates a clear seasonal periodic pattern, and our model performs well in fitting this low-amplitude, high-precision signal, with an RMSE of 0.0236 and an R^2 of 0.9644. The prediction curves for thermal/cooling load are smooth and highly aligned with the actual values, with an RMSE of 1.401 and an R^2 of 0.9583, showcasing the model's stable modeling capability for strongly periodic energy loads. These results serve as a quantitative reference for subsequent ablation studies and baseline analyses.

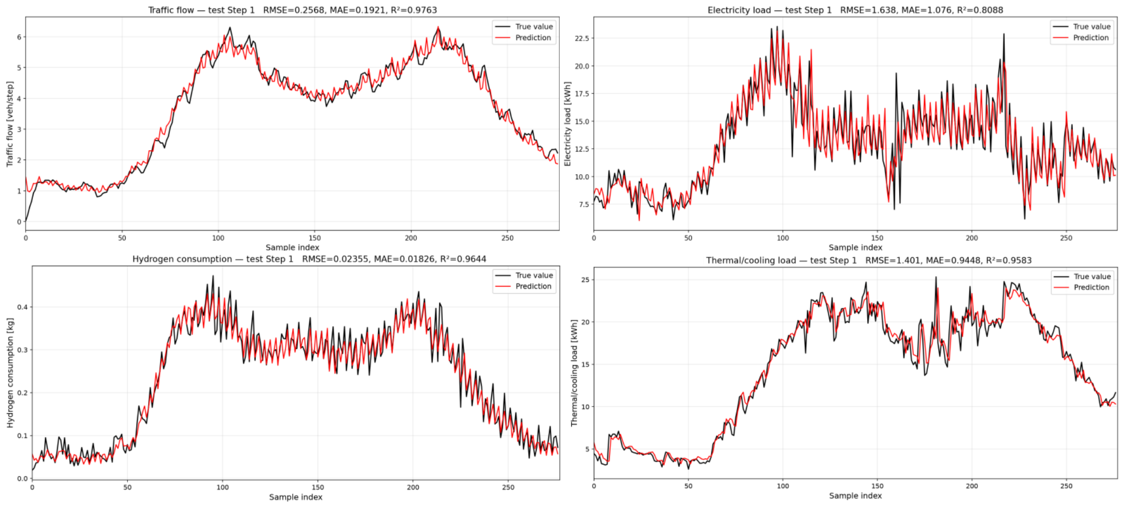

Figure 8. Prediction results of the proposed model on the test set.

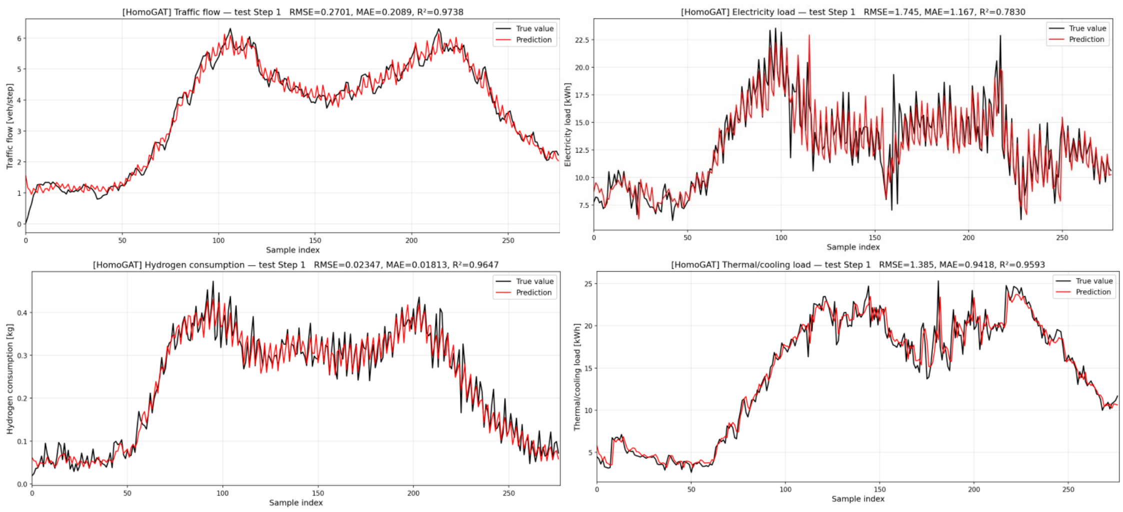

Figure 9. Ablation Experiment a: Homogeneous GAT.

Figure 9 illustrates the prediction performance after replacing HetGAT with a HomoGAT. Compared with Figure 8, it is evident that HomoGAT exhibits significant degradation in traffic flow prediction, with the prediction curve showing weaker amplitude following ability in peak segments. Similarly, there is degradation in power load prediction, indicating that heterogeneous relationships between cross-domain nodes play a crucial role in feature transfer in power load prediction. The equal weight treatment of the homogeneous aggregation strategy leads to the confusion and loss of key cross-domain information. However, HomoGAT's performance in terms of hydrogen consumption and cooling/heating loads is basically on par with, or even slightly better than, that of our model. We analyze this by considering that the heterogeneous coupling between these two types of energy loads and transportation nodes is relatively weak. In this case, the parameter sharing mechanism of the homogeneous graph, which models all nodes uniformly, serves as an implicit regularization effect, reducing the risk of overfitting to some extent with limited training samples. This phenomenon also suggests that the advantage of heterogeneous modeling is most pronounced in cross-domain strongly coupled tasks (such as traffic flow and power load). For tasks with weaker coupling to the transportation side, an adaptive heterogeneous–homogeneous hybrid mechanism can be introduced for further optimization in the future. Overall, the dual heterogeneous mechanism of HetGAT makes an irreplaceable contribution to the two core prediction tasks in this paper, validating the rationality of its design.

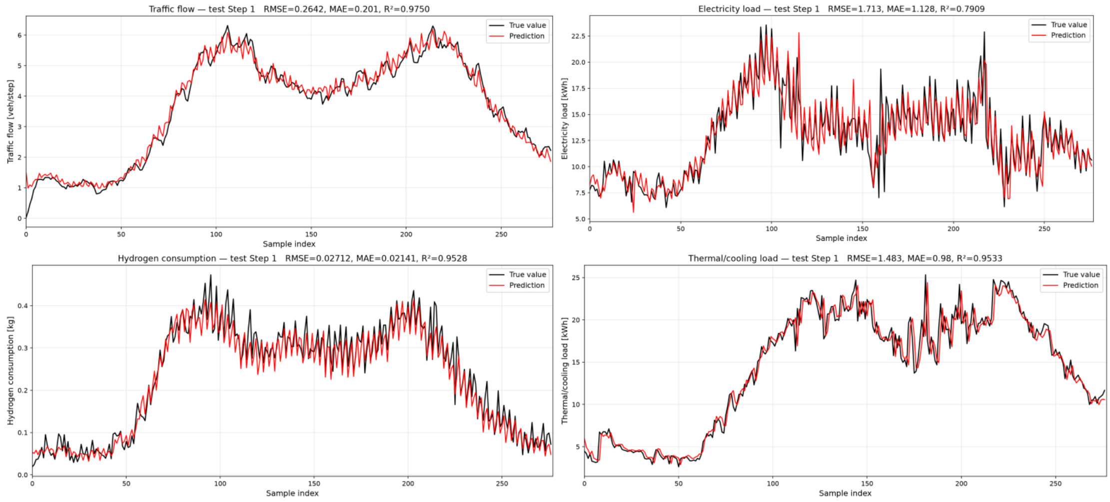

Figure 10. Ablation Experiment b: LSTM.

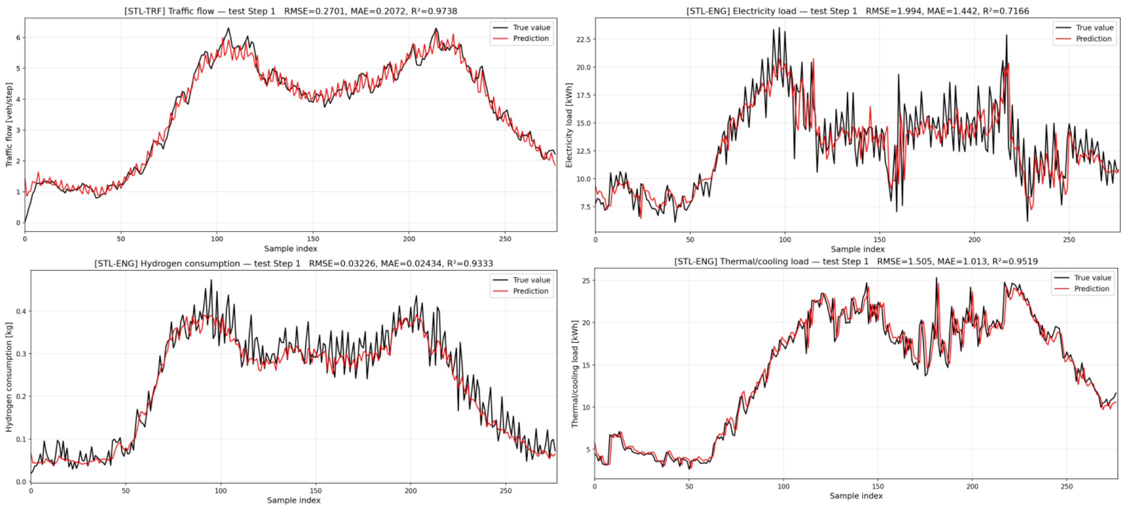

Figure 11. Ablation Experiment c: Single-task.

Figure 10 illustrates the prediction performance after replacing the Transformer temporal encoder with an LSTM. Compared with Figure 8, it can be seen that LSTM is inferior to the complete model of this paper across all four tasks. The above results indicate that the global self-attention mechanism of Transformer exhibits significant advantages in capturing long-range periodic patterns and cross-step dependencies in multi-energy flow time series signals. LSTM processes historical sequences in a step-by-step recursive manner, and its hidden state has inherent limitations due to gradient dispersion during the transmission of long sequences, making it difficult to effectively align the correlation between distant historical periods and the current prediction target. This fully validates the design rationality of Transformer as a time series encoder in this task scenario.

Figure 11 illustrates the prediction performance after splitting the traffic and energy tasks into independent models and eliminating shared backbones for single-task training. Compared with Figure 8, single-task training exhibits the most significant performance degradation in the energy prediction task. The prediction curve shows a notable underestimation during peak hours, and the high-frequency oscillation part of the power load is difficult to effectively track. Hydrogen consumption degradation is the most pronounced, indicating a strong implicit synergistic relationship between hydrogen consumption and traffic flow. In terms of traffic flow, the degradation in single-task training is relatively limited, but still inferior to the accuracy achieved through multi-task joint training. The aforementioned results verify the necessity of the shared backbone multi-task framework from multiple dimensions: the transportation and energy systems are deeply coupled at the physical level, with traffic flow fluctuations directly driving changes in various energy consumption demands such as charging, hydrogen refueling, and heating. Through joint multi-task training, the shared backbone is compelled to extract cross-domain coupled representations that are beneficial to both types of tasks, thus achieving significant positive transfer effects. It significantly surpasses the performance upper limit of single-task independent modeling in various energy prediction tasks.

5.3. Baseline Method Comparison

To verify the effectiveness of the methods proposed in this paper, all methods were evaluated under the same data partitioning and metrics in comparison with the following two types of baselines:

(1) Temporal-only baseline: Transformer-only, which models temporal dependencies while ignoring spatial topology.
(2) Spatiotemporal coupling model: GCN+LSTM (a Graph Convolutional Network (GCN) combined with LSTM), but both adopt homogeneous graphs, do not distinguish node and edge types, and are single-task modeling.

The results are summarized in Table 4, with each row representing a method and each column listing the R^2/RMSE/MAE for traffic and three energy tasks. As summarized in Table 4, the proposed HetGAT–Transformer attains the best or near-best accuracy on both tasks, and its advantage over the homogeneous-graph baseline mainly stems from the explicit modeling of cross-system coupling by the dual-heterogeneous mechanism. The prediction curves are shown in Figures 12 and 13.

Figure 12 illustrates the prediction performance when the graph module is removed and only the Transformer is relied upon for temporal modeling. By comparison, it is evident that Transformer-only significantly underperforms the method proposed in this paper across all four tasks, with the degradation being most pronounced in traffic flow prediction. The prediction curve exhibits notable underestimation and phase shift in the peak region. This result clearly indicates that traffic flow is strongly influenced by the spatial topological relationships among multiple nodes, and the flow propagation characteristics between adjacent nodes in the road network constitute crucial information that cannot be captured solely by relying on temporal attention mechanisms. Overall, the spatial heterograph module is an indispensable core component of the model proposed in this paper, and its explicit modeling capability for topological associations between nodes within the system is the key to surpassing pure temporal methods.

Figure 13 illustrates the prediction performance of single-task modeling using the homogeneous GCN+LSTM. By comparison, it can be seen that the RMSE (0.2516) of GCN+LSTM on the traffic flow task is relatively close to that of our method (0.2468), with a difference of about 1.9%. However, the R^2 (0.9773 vs 0.9763) is slightly higher, which is mainly attributed to the rapid following ability of LSTM for short-term local fluctuations in traffic flow and the basic capture of spatial propagation in the road network by GCN under homogeneous neighborhood aggregation. Nevertheless, GCN+LSTM exhibits significant degradation in the three energy prediction tasks. The aforementioned degradation reveals two key limitations. First, the equal-weight neighborhood aggregation of GCN fails to distinguish the semantic differences between traffic nodes, energy nodes, and cross-system coupling edges, resulting in the dilution of cross-domain energy coupling features during propagation. Second, GCN+LSTM is a single-task independent modeling approach, completely severing the gradient transfer path between traffic flow and energy load, and unable to obtain cross-domain positive transfer benefits from joint training, ultimately leading to a comprehensive decline in energy prediction accuracy. The method proposed in this paper achieves comprehensive superiority over GCN+LSTM in terms of the comprehensive performance of energy prediction through the synergistic effects of HetGAT's dual heterogeneous modeling, Transformer's global temporal encoding, and multi-task shared backbone.

Table 4. Baseline comparison (traffic and energy tasks reported separately; R^2: higher is better; RMSE & MAE: lower is better).

Method	Traffic Flow			Electricity Load			Hydrogen Load			Thermal-Cooling Consumption		
	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE
Transformer-only	0.9613	0.3286	0.2539	0.7859	1.7071	1.2538	0.9223	0.0346	0.0271	0.9491	1.5505	1.0399
GCN+LSTM	0.9773	0.2516	0.1870	0.7193	1.9850	1.3590	0.9259	0.0340	0.0257	0.9477	1.5690	1.0250
Our model	0.9763	0.2468	0.1921	0.8088	1.6380	1.0760	0.9644	0.0236	0.0183	0.9583	1.4010	0.9448

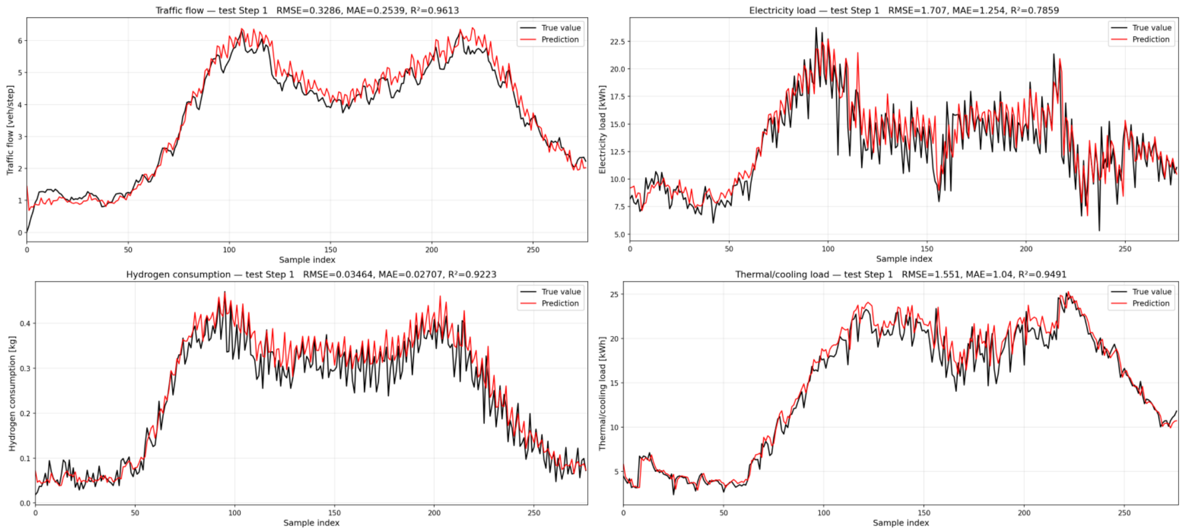

Figure 12. Baseline method 1: Only Transformer.

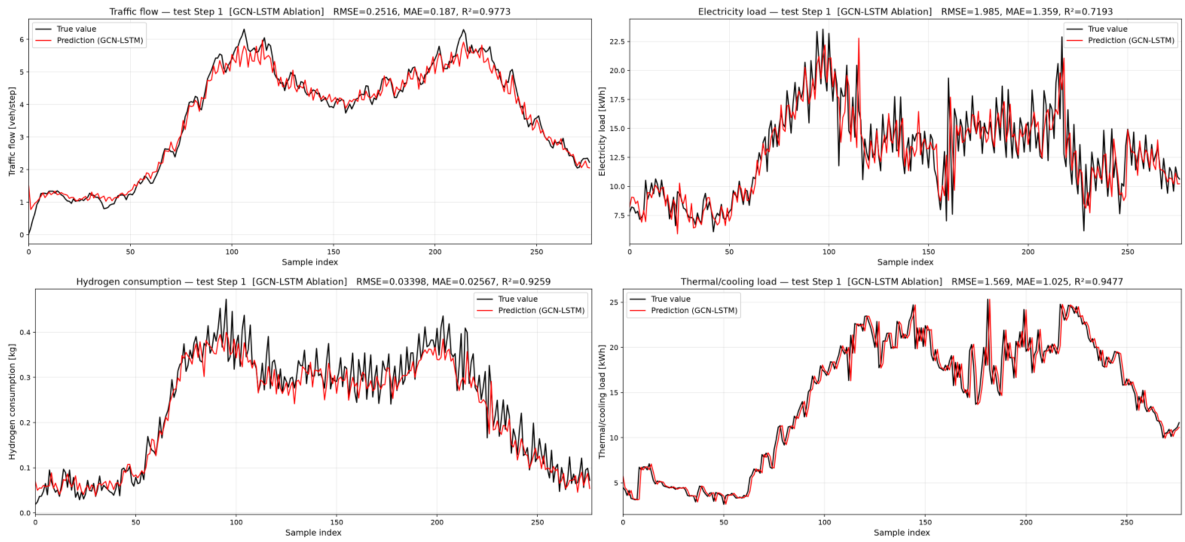

Figure 13. Baseline method 2: GCN+LSTM.

6. Conclusions

This paper studied the coordinated prediction of traffic flows and multi-energy flows in large-scale transportation hubs, where multi-source data are hard to unify and the traffic and energy subsystems are tightly coupled in space and time. To resolve these two difficulties, a spatiotemporal coupled prediction method that combines an H3-grid unified representation with a HetGAT–Transformer hybrid model was proposed. Multi-source data are first mapped onto H3 hexagonal grids and fused into a standardized fourth-order tensor through area, length, and frequency spatial weights. A heterogeneous graph with four node types and five edge types then represents the spatial topology, over which a two-layer HetGAT and a Transformer encoder jointly model spatial coupling and long-range temporal evolution, and an uncertainty-weighted dual-task head yields traffic and energy forecasts within a single network.

On the SUMO-based Ningbo-Zhoushan Port scenario, the proposed method reached coefficients of determination of 0.9763, 0.8088, 0.9644, and 0.9583 for traffic flow, electricity load, hydrogen consumption, and thermal/cooling load, respectively. The comparatively lower R^2 for electricity-load prediction mainly results from its stronger high-frequency variability and the superposition of mobile and fixed electricity demands, which makes this task less regular than hydrogen and thermal/cooling load forecasting.The ablation study confirmed the contribution of each component. Degrading the heterogeneous graph to a homogeneous GAT lowered the electricity-load R^2 from 0.8088 to 0.7830, replacing the Transformer with an LSTM weakened all four tasks, and splitting the two tasks into independent single-task models caused the largest drop on the energy side, with the hydrogen R^2 falling from 0.9644 to 0.9333. Compared with the Transformer-only and GCN+LSTM baselines, the method obtained the lowest errors on the three energy tasks while remaining comparable on traffic flow, which indicates that jointly modeling heterogeneous spatial topology, long-range temporal dependencies, and cross-task coupling is the key to accurate multi-energy flow prediction.

The present study relies on simulation data, whose distribution may differ from field measurements. Future work will validate the method on real operational data from transportation hubs and investigate cross-regional transfer. In particular, transferring the model across ports or other transportation hubs is challenging because road-network topology, H3-grid occupancy, energy-facility density, energy-carrier composition, and traffic–energy demand distributions may differ substantially across regions, resulting in both feature-distribution and graph-topology shifts. Future research will therefore explore topology-aware domain adaptation and transferable heterogeneous representations, together with privacy-preserving training, to support practical online scheduling and coordinated optimization of multi-energy flows.

Author Contributions

W.J.: conceptualization, methodology, software, writing—original draft preparation; L.Z.: supervision; Z.Z.: funding acquisition, project administration, writing—reviewing and editing; D.Z.: data curation, validation, visualization, investigation; Q.C.: funding acquisition, writing—reviewing and editing. All authors have read and agreed to the published version of the manuscript.

Funding

This work was supported by the National Key Research and Development Program of China under Grant 2024YFB2606304, the Natural Science Foundation of Hubei Province under Grants 2026AFA011 and 2025AFB188, and the Natural Science Foundation Exploration Program (Chenguang Program) of Wuhan under Grant 2025040601020125.

Institutional Review Board Statement

Not applicable.

Informed Consent Statement

Not applicable.

Data Availability Statement

Not applicable.

Acknowledgments

The authors would like to thank the School of New Energy and Electrical Engineering, Wuhan University of Technology, for the research support provided for this work.

Conflicts of Interest

All authors declare no conflict of interest.

Use of AI and AI-Assisted Technologies

No AI tools were utilized for this paper.

References

1. Chen, S.; Wei, Z.; Gu, W.; et al. Energy System Transition and Transformation toward Carbon Neutrality: Multi-Energy Flow Coordination Technologies. *Electr. Power Autom. Equip.* **2021**, *41*, 3–12. https://doi.org/10.16081/j.epae.202109037. (In Chinese)
2. Guo, Y.; Li, Y.; Zhou, S.; et al. Optimal Dispatch for Integrated Energy System Considering Data-Driven Dynamic Energy Hubs and Thermal Dynamics of Pipeline Networks. *IEEE Trans. Smart Grid* **2024**, *15*, 4537–4549. https://doi.org/10.1109/tsg.2024.3382740.
3. Qian, T.; Liang, Z.; Shao, C.; et al. Unsupervised Learning for Efficiently Distributing EVs Charging Loads and Traffic Flows in Coupled Power and Transportation Systems. *Appl. Energy* **2025**, *377*, 124476. https://doi.org/10.1016/j.apenergy.2024.124476.
4. Li, J.; Zhu, M.; Lu, Y.; et al. Review on Optimal Scheduling of Integrated Energy Systems. *Power Syst. Technol.* **2021**, *45*, 2256–2269. https://doi.org/10.13335/j.1000-3673.pst.2021.0020. (In Chinese)

5. Geidl, M.; Koeppel, G.; Favre-Perrod, P.; et al. The Energy Hub—A Powerful Concept for Future Energy Systems. In Proceedings of the Third Annual Carnegie Mellon Conference on the Electricity Industry, Pittsburgh, PA, USA, 13–14 March 2007.
6. Mohammadi, M.; Noorollahi, Y.; Mohammadi-Ivatloo, B.; et al. Energy Hub: From a Model to a Concept—A Review. *Renew. Sustain. Energy Rev.* **2017**, *80*, 1512–1527. https://doi.org/10.1016/j.rser.2017.07.030.
7. Song, C.; Yang, H.; Cai, J.; et al. Multi-Energy Load Forecasting via Hierarchical Multi-Task Learning and Spatiotemporal Attention. *Appl. Energy* **2024**, *373*, 123788. https://doi.org/10.1016/j.apenergy.2024.123788.
8. Su, Z.; Zheng, G.; Hu, M.; et al. Short-Term Load Forecasting of Regional Integrated Energy System Based on Spatio-Temporal Convolutional Graph Neural Network. *Electr. Power Syst. Res.* **2024**, *232*, 110427. https://doi.org/10.1016/j.epsr.2024.110427.
9. Kim, H.J.; Kim, D.; Tak, H.; et al. Global-Local Attention-Enabled Multiple Decoder Transformer for Multi-Energy Load Forecasting in User-Level Integrated Energy System. *Appl. Energy* **2025**, *396*, 126255. https://doi.org/10.1016/j.apenergy.2025.126255.
10. Yang, X.; Yun, J.; Zhou, S.; et al. A Spatiotemporal Distribution Prediction Model for Electric Vehicles Charging Load in Transportation Power Coupled Network. *Sci. Rep.* **2025**, *15*, 4022. https://doi.org/10.1038/s41598-025-88607-y.
11. Scarselli, F.; Gori, M.; Tsoi, A.C.; et al. The Graph Neural Network Model. *IEEE Trans. Neural Netw.* **2009**, *20*, 61–80. https://doi.org/10.1109/tnn.2008.2005605.
12. Jin, C.; Ruan, T.; Wu, D.; et al. HetGAT: A Heterogeneous Graph Attention Network for Freeway Traffic Speed Prediction. *J. Ambient Intell. Humaniz. Comput.* **2021**. https://doi.org/10.1007/s12652-020-02807-0.
13. Vaswani, A.; Shazeer, N.; Parmar, N.; et al. Attention Is All You Need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
14. Li, B.; Li, Z.; Chen, J.; et al. MAST-GNN: A Multimodal Adaptive Spatio-Temporal Graph Neural Network for Airspace Complexity Prediction. *Transp. Res. Part C Emerg. Technol.* **2024**, *160*, 104521. https://doi.org/10.1016/j.trc.2024.104521.
15. Hou, D.; Gao, L.; Qian, Z.; et al. Vehicle Model Development and Integration for a Heterogeneous Traffic Flow Simulation Platform Based on SUMO Software. *J. Tongji Univ. (Nat. Sci.)* **2021**, *49*, 957–963. https://doi.org/10.11908/j.issn.0253-374x.21156. (In Chinese)
16. Gallardo-Gómez, J.A.; Díaz-del-Río, F.; Morón Fernández, M.J.; et al. An Iterative Calibration and Validation Methodology for Large-Scale Traffic Simulations in SUMO. In Proceedings of the 40th ACM SIGSIM International Conference on Principles of Advanced Discrete Simulation, Vienna, Austria, 24–26 June 2026; pp. 197–198. https://doi.org/10.1145/3806789.3810970.
17. Khan, M.A.; Jang, J.H.; Iqbal, N.; et al. Enhancing Patient Rehabilitation Predictions with a Hybrid Anomaly Detection Model: Density-Based Clustering and Interquartile Range Methods. *CAAI Trans. Intell. Technol.* **2025**, *10*, 983–1006. https://doi.org/10.1049/cit2.70000.
18. Wu, B.; Chen, Z.; Lu, X.; et al. A Hierarchical Model with Hexagon Grids for Multi-Objective Route Planning in Large-Scale Off-Road Environments. *Int. J. Geogr. Inf. Sci.* **2024**, *38*, 1388–1413. https://doi.org/10.1080/13658816.2024.2347319.
19. Liu, R.; Wang, Z.; Han, T.; et al. HetGAT-LMI: Interpretable Heterogeneous Graph Attention Method for Predicting lncRNA–miRNA Interactions. *J. Chem. Inf. Model.* **2026**, *66*, 796–807. https://doi.org/10.1021/acs.jcim.5c02664.
20. Veličković, P.; Cucurull, G.; Casanova, A.; et al. Graph Attention Networks. In Proceedings of the International Conference on Learning Representations (ICLR), Vancouver, BC, Canada, 30 April–3 May 2018. https://doi.org/10.48550/arxiv.1710.10903.
21. Ba, J.L.; Kiros, J.R.; Hinton, G.E. Layer Normalization. *arXiv* **2016**, arXiv:1607.06450.
22. He, K.; Zhang, X.; Ren, S.; et al. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. https://doi.org/10.1109/cvpr.2016.90.
23. Kendall, A.; Gal, Y.; Cipolla, R. Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 7482–7491. https://doi.org/10.1109/cvpr.2018.00781.