EAI Endorsed Transactions
on Energy Web
Research Article EAI.EU

Big Data-Driven Short-Term Load Forecasting in Smart Grids: A Spatio-Temporal Dynamic Graph Transformer with Uncertainty-Aware Attention

Jing Zhou[1], Jiyue Long[2], and Xiaohong Long[1,*]

[1]School of Artificial Intelligence & Big Data, Luzhou Vocational & Technical College, Luzhou, 646000, Sichuan, China
[2]School of Electrical and Electronic Engineering, Huazhong University of Science and Technology, Wuhan, 430000, Hubei, China

Abstract

Short-term load forecasting for modern smart grids must jointly model long-range temporal patterns, correlations across many metering points, and the uncertainty required for operational decision-making. We propose ST-DGT-UA, a spatio-temporal dynamic graph Transformer that (i) embeds heterogeneous covariates with convolutional projections and Time2Vec, (ii) learns a time-varying adjacency matrix from node representations, (iii) couples spatial graph attention with temporal ProbSparse attention for efficient long-sequence modeling, and (iv) outputs multiple conditional quantiles trained with multi-quantile Pinball loss. Experiments on GEFCom2014 and the UCI Electricity Load Diagrams datasets show that ST-DGT-UA achieves MAPE/RMSE of 1.92/145.6 on GEFCom2014 and 2.65/42.3 on UCI, and improves probabilistic quality with Pinball Loss 0.024 and CRPS 0.043. These results indicate that learning dynamic, data-driven spatial dependencies is beneficial for multi-node load forecasting where correlations evolve over time.

Received on 20 December 2025; accepted on 20 March 2026; published on 11 August 2026

Keywords: short-term load forecasting, smart grid, dynamic graph neural network, Transformer, uncertainty quantification, deep learning, spatio-temporal data mining

doi:10.4108/ew.13839

1. Introduction

Short-term load forecasting (STLF) underpins day-ahead scheduling, reserve planning, and electricity market operations in modern power systems. With higher penetration of weather-dependent renewables and fast-growing flexible loads (e.g., electric vehicle charging), the net load becomes more volatile and the uncertainty faced by dispatchers increases [1]. Meanwhile, AMI and WAMS continuously stream high-frequency measurements for many metering points, together with meteorological covariates, turning STLF into a high-dimensional spatio-temporal learning problem [2].

In this setting, three issues repeatedly limit forecasting accuracy. First, treating each node as an independent time series (as in many ARIMA/LSTM formulations) discards cross-node interactions and often underfits multi-node datasets [3]. Second, most graph-based approaches rely on static adjacency matrices derived from physical topology or geographic proximity, although operational coupling and weather-driven correlations can vary over time [4]. Third, higher sampling rates and longer input windows increase sequence length, so RNN-based models may suffer from degraded memory and training inefficiency, while vanilla self-attention incurs $O(L^2)$ cost [5]. Finally, for risk-aware operation, point forecasts alone are insufficient; practical applications require calibrated uncertainty estimates rather than post-hoc intervals [6].

Motivated by these constraints, we propose ST-DGT-UA, an end-to-end framework that jointly learns

*Corresponding author. Email: longxiaohong@lzy.edu.cn

dynamic spatial dependencies, models long-range temporal patterns efficiently, and outputs multiple quantiles for probabilistic forecasting. The main contributions of this work are summarized as follows:

- We design a metric-learning graph generator that produces a time-varying adjacency matrix from node representations, enabling the model to capture implicit correlations beyond static physical topology priors.
- We couple spatial graph attention with temporal ProbSparse attention to model multi-node long sequences with reduced attention complexity while preserving forecasting accuracy.
- We formulate a multi-quantile regression trained with a multi-quantile Pinball loss so that the model directly outputs prediction intervals together with the median forecast.
- We integrate convolutional projections for numerical covariates and Time2Vec-based temporal encodings to represent periodic calendar effects and external factors in a unified latent space.
- We evaluate the proposed method on GEFCom2014 and UCI Electricity Load Diagrams, reporting both deterministic errors (MAPE/RMSE) and probabilistic scores (Pinball Loss/CRPS) against strong baselines.

The remainder of this paper is organized as follows. Section 2 systematically reviews related literature in the short-term load forecasting domain. Section 3 elaborates on the theoretical framework, mathematical principles, and core module design of the ST-DGT-UA model. Section 4 introduces the experimental setup, including dataset characteristics, baseline model selection, and evaluation metric definitions. Section 5 provides multi-dimensional in-depth analysis and discussion of experimental results. Section 7 summarizes the paper and outlines future research directions.

2. Related Work

Short-term load forecasting (STLF) has evolved from statistical models to modern deep learning methods, driven by richer sensing data and the need to model nonlinear dynamics and cross-node dependencies. This section reviews work most relevant to our setting—multi-node spatio-temporal forecasting with long input windows and uncertainty estimation—covering statistical/ML baselines, deep sequence models, graph neural networks for spatial modeling, Transformer-based long-sequence models, and probabilistic forecasting approaches.

2.1. Traditional Statistical and Machine Learning Methods

Early STLF research primarily relied on linear time series analysis, with representative methods including Autoregressive Integrated Moving Average (ARIMA), Exponential Smoothing, and Kalman Filtering. These methods possess rigorous mathematical foundations and low computational overhead, performing well during periods of stable load fluctuation. However, they are inherently linear models that struggle to capture the nonlinear characteristics and abrupt changes commonly present in load data [2].

To overcome linear limitations, machine learning methods such as Support Vector Regression (SVR), Random Forest (RF), and Gradient Boosting Decision Trees (GBDT/XGBoost/LightGBM) have been widely applied. These methods significantly enhance nonlinear fitting capability through kernel tricks or ensemble strategies [2]. SVR, in particular, can effectively handle small-sample problems by mapping data to high-dimensional spaces. However, the performance of machine learning methods highly depends on the quality of manual feature engineering, such as manually constructing lag features and sliding window statistics. When processing high-dimensional spatio-temporal data, feature dimension explosion often leads to the curse of dimensionality.

2.2. Deep Learning Approaches

Since 2015, deep learning has gradually dominated STLF due to its end-to-end feature extraction capabilities. Recurrent models built on LSTM [7] and GRU remain widely used for load forecasting, including residential STLF with LSTM-based architectures [3]. To better capture nonlinearities and local temporal patterns, convolutional and residual structures have also been explored, e.g., deep residual networks for short-term load forecasting [4]. Beyond point forecasts, probabilistic variants based on quantile regression have been studied to provide calibrated uncertainty for operation, often trained with Pinball loss [6, 8].

Nevertheless, RNN-class models possess two inherent deficiencies. First, their sequential computation mechanism limits training efficiency. Second, although LSTM improves memory capability, its ability to capture global dependencies remains limited when facing ultra-long sequences with thousands of time steps. The serial nature of these architectures prevents full utilization of GPU parallel computing power, resulting in inefficient training processes.

2.3. Graph Neural Networks for Spatial Modeling

Power systems inherently possess graph-structured properties. The introduction of Graph Neural Networks (GNNs) has extended load forecasting from single temporal dimension analysis to spatio-temporal dimensions. Early GNN research primarily relied on Spectral Graph Convolution using eigendecomposition of the Laplacian matrix for convolution operations. Most studies assume static graph structures, typically constructing adjacency matrices based on geographic distance (K-NN graphs) or physical connections (transmission lines). GraphSAGE [9], for instance, improves training efficiency on large-scale graphs through neighbor node sampling for information aggregation.

However, physical connections do not fully equate to load feature correlations, and spatial dependencies can evolve with weather conditions and operating states. Recent work therefore, leverages data-driven graph learning for STLF, for example, spatial-temporal residential STLF via graph neural networks [10] and spatial-temporal embedding graph neural networks [11]. Multi-scale adaptive graph learning [12] and comprehensive spatial-temporal modeling [13] further advance representation learning for time series forecasting. To handle nonstationary correlations, dynamic/adaptive graph construction mechanisms have been proposed for load forecasting [14, 15], while multi-area STLF has also been explored with spatiotemporal GNNs [16]. These findings motivate learning time-varying spatial dependencies rather than relying solely on static topology priors.

2.4. Transformer Architectures for Time Series

The emergence of the Transformer architecture [17] has fundamentally transformed the paradigm of sequence modeling. Its full self-attention mechanism allows models to directly establish dependencies between any two points in a sequence, regardless of distance. Several Transformer variants have been proposed for time series forecasting to address its unique characteristics.

The iTransformer model [5] inverts the Transformer architecture by applying attention across variate tokens rather than time points, effectively capturing multivariate correlations for time series forecasting. TimesNet [18] transforms 1D time series into 2D tensors based on multiple periods, enabling temporal 2D-variation modeling for general time series analysis. PatchTST [19] aggregates temporal points into patches, not only preserving local semantics but also significantly reducing sequence length. FEDformer [20] further improves long-sequence forecasting by combining decomposition with frequency-domain enhancement. In the energy domain, Transformer-based approaches have been applied to electricity load forecasting [21].

Although these models perform excellently on general time series prediction datasets, in power load forecasting, if multi-node data is directly treated as multivariate sequences while ignoring spatial topological structure, optimal performance often cannot be achieved. How to organically combine Transformer's efficient temporal modeling capability with GNN's spatial aggregation capability represents a current research hotspot.

2.5. Probabilistic Forecasting and Uncertainty Quantification

With increasing uncertainty in smart grids, the importance of probabilistic load forecasting (PLF) has become increasingly prominent. Mainstream PLF methods include non-parametric quantile regression, which directly optimizes Pinball Loss to predict specific quantiles [6, 8, 22], hybrid probabilistic forecasting approaches for distribution networks [23], and Bayesian deep learning that estimates uncertainty through various inference techniques [24]. Recent studies also explore attention-enhanced deep architectures for short-term load forecasting [25].

Existing literature predominantly treats probabilistic forecasting as an independent post-processing step or focuses solely on univariate probability distributions, lacking end-to-end probabilistic forecasting frameworks that integrate dynamic spatio-temporal features. This gap motivates the development of unified frameworks that can simultaneously address deterministic and probabilistic forecasting objectives.

2.6. Research Gap Analysis

Synthesizing the above analysis, current research primarily exhibits the following gaps. First, dynamic modeling remains insufficient, as most GNN models still adhere to static graphs without reflecting changes in real-time grid operational states and implicit correlations. Second, spatio-temporal decoupling persists, where Transformers excel at temporal modeling and GNNs excel at spatial modeling, but simple serial combinations such as GCN-LSTM struggle to achieve deep spatio-temporal feature fusion, with computational efficiency often constrained by RNN components. Third, deterministic and probabilistic aspects remain separated, as high-precision deep learning models mostly focus on point prediction while lacking embedded, low-computational-cost uncertainty quantification mechanisms.

The ST-DGT-UA model proposed in this paper is designed precisely to fill these gaps. Through deep integration of dynamic graph learning with ProbSparse Transformer, and incorporating multi-quantile regression at the output end, we achieve

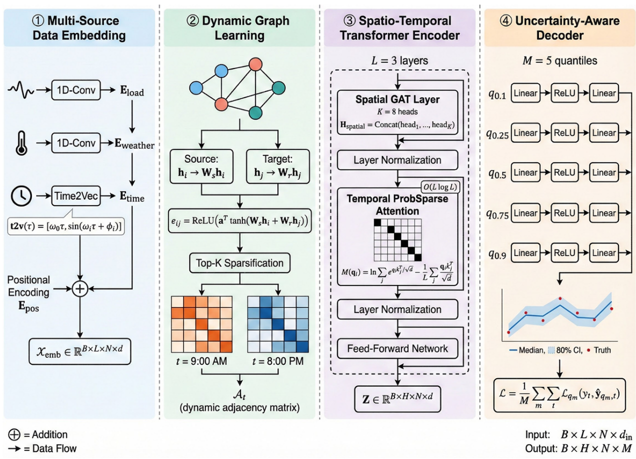

Figure 1. Overall architecture of the proposed ST-DGT-UA framework, comprising four core components: (a) Multi-Source Data Embedding Layer, (b) Dynamic Graph Learning Module, (c) Spatio-Temporal Transformer Encoder, and (d) Uncertainty-Aware Decoder

high-accuracy, high-efficiency load forecasting with risk quantification capability.

3. Methodology

This section elaborates on the mathematical principles and implementation details of the ST-DGT-UA model. The overall architecture is illustrated in Fig. 1, comprising four core components: the Multi-Source Data Embedding Layer, the Dynamic Graph Learning Module, the Spatio-Temporal Transformer Encoder, and the Uncertainty-Aware Decoder.

3.1. Problem Formulation

Consider a power network with N nodes, which may represent substations, smart meters, or regional load aggregation points. At time t, the observed feature matrix is denoted as $\mathbf{X}_t \in \mathbb{R}^{N\times D}$, where D represents the feature dimension including load values, temperature, humidity, and temporal encodings.

Given a historical time window of length L with observation sequence $\mathcal{X}_{hist} = \{\mathbf{X}_{t-L+1}, \dots, \mathbf{X}_t\}$, our objective is to predict load values for future H time steps $\mathcal{Y}_{pred} = \{\mathbf{Y}_{t+1}, \dots, \mathbf{Y}_{t+H}\}$. Unlike point prediction, we aim to estimate the conditional quantile distribution:

$$\hat{q}_\tau(\mathbf{Y}_{t+h}|\mathcal{X}_{hist}) \tag{1}$$

where $\tau \in (0, 1)$ denotes the quantile level, such as 0.1, 0.5, or 0.9.

3.2. Multi-Source Data Embedding Layer

Input data are heterogeneous with different physical meanings, and direct input to neural networks yields suboptimal results. We therefore map them to a high-dimensional latent space through specialized embedding mechanisms.

For continuous numerical variables such as load values, temperature, and wind speed, we employ one-dimensional convolution (1D-Conv) layers to project them to the model dimension d_{model}:

$$\mathbf{E}_{val} = \text{Conv1D}(\mathbf{X}_{numeric}) \tag{2}$$

Temporal features such as hour-of-day and day-of-week exhibit strong periodicity. Traditional one-hot encoding leads to dimension explosion and cannot reflect periodic continuity, as 23:00 and 00:00 are actually adjacent. Following recent advances in temporal encoding for time series Transformers [26], we adopt a learnable periodic temporal embedding inspired by Fourier transforms, which can capture temporal invariance and periodicity. The temporal embedding is defined as:

$$\mathbf{TE}(\tau)[i] = \begin{cases} \omega_i\tau + \phi_i, & \text{if } i = 0 \\ \sin(\omega_i\tau + \phi_i), & \text{if } 1 \le i \le k \end{cases} \tag{3}$$

where τ is the scalar time input, and ω_i and ϕ_i are learnable frequency and phase parameters. Through combinations of different frequencies ω_i, the model can adaptively learn daily, weekly, and even seasonal cycles.

The final embedding representation sums all components:

$$\mathcal{X}_{emb} = \mathbf{E}_{val} + \mathbf{E}_{time} + \mathbf{E}_{pos} \tag{4}$$

where $\mathbf{E}_{pos}$ is the standard Transformer positional encoding for preserving sequence order information. At this stage, $\mathcal{X}_{emb} \in \mathbb{R}^{B\times L\times N\times d_{model}}$.

3.3. Dynamic Graph Learning Module

This module distinguishes our model from traditional GNNs. To capture spatially evolving dependencies over time, we no longer input a fixed adjacency matrix A but instead adaptively generate a dynamic adjacency matrix $\mathcal{A}_t$ through node embedding similarity.

Inspired by recent advances in adaptive graph neural networks for spatio-temporal modeling [12, 13], we design a metric learning-based graph generator. Suppose at time step t, the feature vectors of nodes i and j are $h_i^{(t)}$ and $h_j^{(t)}$ respectively. We first transform node features through two learnable linear transformation weights $\mathbf{W}_s$ and $\mathbf{W}_r$, representing source and target node information spaces respectively. The connection weight $e_{ij}^{(t)}$ between nodes is then computed as:

$$e_{ij}^{(t)} = \text{ReLU}\left(\mathbf{a}^T \cdot \tanh\left(\mathbf{W}_s h_i^{(t)} + \mathbf{W}_r h_j^{(t)}\right)\right) \tag{5}$$

where $\mathbf{a}$ is the attention vector. To sparsify the graph structure for noise reduction and computational efficiency improvement, we introduce a Top-K strategy that retains only the K neighbors with the largest weights, setting the rest to zero.

Finally, the softmax function normalizes the weights to obtain the dynamic adjacency matrix:

$$\mathcal{A}_{ij}^{(t)} = \frac{\exp(e_{ij}^{(t)})}{\sum_{k \in \mathcal{N}_i} \exp(e_{ik}^{(t)})} \tag{6}$$

The resulting $\mathcal{A}_t$ is an asymmetric, time-varying matrix that precisely characterizes instantaneous energy flows or fault propagation paths in the power grid. For instance, when load surges in a certain region, the connection weight with upstream substations automatically increases.

3.4. Spatio-Temporal Transformer Encoder

This module aims for deep fusion of spatio-temporal features through alternating stacking of Spatial Graph Attention Layers and Temporal ProbSparse Attention Layers.

Spatial Graph Attention Layer. In this layer, we utilize the dynamic adjacency matrix $\mathcal{A}_t$ generated in the previous subsection for message passing. Unlike standard GCN's isotropic aggregation, we incorporate Multi-Head Graph Attention mechanisms. For each attention head k, the spatial aggregation feature of node i is computed as:

$$H_{spatial,i}^{(t,k)} = \sigma\left(\sum_{j \in \mathcal{N}_i} \mathcal{A}_{ij}^{(t)} \mathbf{W}^k h_j^{(t)}\right) \tag{7}$$

The final output concatenates multi-head features:

$$H_{spatial}^{(t)} = \text{Concat}_{k=1}^{K}(H_{spatial}^{(t,k)}) \tag{8}$$

This step achieves spatial dimension information aggregation, enabling each node's features to incorporate not only its own history but also the real-time states of correlated nodes.

Temporal ProbSparse Attention Layer. To address the $O(L^2)$ computational bottleneck of traditional Self-Attention, we adopt ProbSparse Self-Attention [5, 26]. Due to the long-tail distribution characteristics of load data, self-attention maps are typically sparse, with only a small number of queries having high responses to specific keys. The ProbSparse mechanism uses Kullback-Leibler divergence to measure the difference between query distribution and uniform distribution.

The sparsity measure $M(q_i, K)$ for the i-th query is defined as:

$$M(q_i, K) = \ln \sum_{j=1}^{L} e^{\frac{q_i k_j^T}{\sqrt{d}}} - \frac{1}{L} \sum_{j=1}^{L} \frac{q_i k_j^T}{\sqrt{d}} \tag{9}$$

We select only the Top-u queries with largest M values (termed Active Queries) for dot-product computation, replacing the remaining queries with mean values. This reduces computational complexity to $O(L \log L)$:

$$\text{ProbSparse}(Q, K, V) = \text{Softmax}\left(\frac{Q_{active} K^T}{\sqrt{d}}\right) V \tag{10}$$

This mechanism enables the model to efficiently process historical sequences exceeding 720 hours (one month) or longer, thereby capturing monthly or seasonal long-range dependencies.

3.5. Uncertainty-Aware Decoder

To quantify prediction uncertainty, the model output is no longer a single value but a set of quantiles for load at future time $t + \tau$. Let the target quantile set be $Q = \{q_1, q_2, \ldots, q_M\}$, for example $\{0.1, 0.25, 0.5, 0.75, 0.9\}$. The model output layer contains M independent linear regression heads, each corresponding to a quantile.

Training employs Multi-Quantile Pinball Loss [6, 8, 22]. For a single quantile τ, Pinball Loss is defined as:

$$L_\tau(y, \hat{y}) = \begin{cases} \tau(y - \hat{y}), & \text{if } y \geq \hat{y} \\ (1 - \tau)(\hat{y} - y), & \text{if } y < \hat{y} \end{cases} \tag{11}$$

The geometric interpretation of this loss function is that when predicted values are too low, the penalty weight is τ; when predicted values are too high, the penalty weight is $1 - \tau$. The total loss function averages all target quantile losses:

$$\mathcal{L}_{total} = \frac{1}{M} \sum_{m=1}^{M} \sum_{t=1}^{T} L_{q_m}(y_{true}^{(t)}, \hat{y}_{q_m}^{(t)}) \tag{12}$$

Through this approach, the model learns not only to predict the median ($q = 0.5$, the most probable load value) but also distribution boundaries (such as $q = 0.1$ and $q = 0.9$), thereby constructing confidence intervals.

4. Experiments

4.1. Datasets

To comprehensively validate model performance and generalization capability, we selected two of the most representative public datasets in the load forecasting domain.

The GEFCom2014 (Global Energy Forecasting Competition 2014) dataset [27] originates from the probabilistic load forecasting track of the 2014 Global Energy Forecasting Competition, containing multi-year hourly load data from a U.S. utility company along with corresponding temperature data from 11 weather stations. The time span extends nine years, providing not only load values but also complete historical temperature records, serving as the gold standard for validating model sensitivity to meteorological factors and probabilistic forecasting capability. We designated the final 12 months of data as the test set, with preceding data split 8:2 into training and validation sets. Missing values were filled using linear interpolation, and numerical values underwent Z-score standardization.

The UCI Electricity Load Diagrams dataset [28] contains 15-minute electricity consumption records for 370 users spanning 2011–2014. This represents a typical high-dimensional, multi-node dataset. With 370 nodes included, it is particularly suitable for validating model spatial aggregation capability and dynamic graph learning capability on large-scale networks. For consistency with GEFCom2014, we resampled data to hourly granularity and applied Z-score standardization.

4.2. Baselines and Evaluation Metrics

To demonstrate the advancement of our model, we compare ST-DGT-UA against baselines spanning four categories. Traditional statistical and machine learning models include ARIMA as a classical univariate time series model serving as a linear baseline, and SVR representing traditional machine learning nonlinear models. Deep sequential models include multi-layer LSTM representing RNN-series state-of-the-art, TCN [25] representing CNN-series state-of-the-art, and N-BEATS [29] based on pure MLP architecture with outstanding performance in the M4 competition. Transformer variants include Informer [5] specifically optimized for long sequences and Autoformer [18] based on series decomposition for capturing periodicity. Spatio-temporal graph models include GCN-LSTM as a simple serial combination of static graph convolution [30] and LSTM, and GraphSAGE [9] as an inductive graph representation learning method for large-scale graph data.

We employ both deterministic and probabilistic metrics for comprehensive evaluation. For deterministic metrics evaluating median prediction ($q = 0.5$) accuracy, we use Mean Absolute Percentage Error (MAPE) measuring relative error as $\frac{1}{n}\sum|\frac{y-\hat{y}}{y}| \times 100\%$, and Root Mean Square Error (RMSE) measuring error fluctuation amplitude with higher sensitivity to large errors. For probabilistic metrics evaluating distribution prediction quality, we use Pinball Loss as defined previously, and Continuous Ranked Probability Score (CRPS) [31] measuring the integral distance between predicted distribution cumulative distribution function F and the step function of true observation y:

$$\mathrm{CRPS}(F, y) = \int_{-\infty}^{\infty} (F(z) - \mathbb{I}\{y \le z\})^2 dz \tag{13}$$

Table 1. Deterministic prediction performance comparison across models on GEFCom2014 and UCI Electricity datasets

Category	Model	GEFCom2014		UCI Electricity	
		MAPE	RMSE	MAPE	RMSE
Statistical	ARIMA	5.82	420.5	8.45	102.3
	SVR	4.95	380.2	7.12	95.6
Deep Seq.	LSTM	3.56	280.1	5.12	78.4
	TCN	3.42	275.4	4.98	76.2
	N-BEATS	3.15	255.8	4.50	70.1
Transformer	Informer	2.85	210.4	4.05	65.2
	Autoformer	2.65	198.2	3.88	61.5
GNNs	GCN-LSTM	3.10	245.3	3.50	58.1
	GraphSAGE	3.05	240.1	3.42	57.5
Proposed	ST-DGT-UA	**1.92**	**145.6**	**2.65**	**42.3**

CRPS is the most rigorous metric in probabilistic forecasting, simultaneously rewarding accuracy and sharpness. We also employ the Diebold-Mariano (DM) Test [32] for statistical testing of whether prediction results from two models exhibit significant differences, with p-values below 0.05 indicating rejection of the null hypothesis and significant performance differences.

4.3. Implementation Details

All experiments were conducted on a workstation configured with an NVIDIA RTX 3090 (24GB) GPU using the PyTorch 1.10 deep learning framework. The hyperparameter configuration includes input sequence length of 96 time steps (4 days), prediction horizon of 24 time steps (1 day), model dimension d_{model} of 512, 8 attention heads, 3 encoder layers, dropout rate of 0.1, Adam optimizer with initial learning rate of 1×10^{-4} using cosine annealing decay, batch size of 32, and early stopping when validation loss fails to decrease for 10 consecutive epochs.

5. Results and Analysis

5.1. Main Results

Table 1 presents the point prediction performance comparison of all models on GEFCom2014 and the UCI Electricity datasets.

All deep learning models substantially outperform ARIMA and SVR, demonstrating the absolute advantage of neural networks in capturing nonlinear load dynamics. Informer and Autoformer outperform LSTM and TCN, particularly on the GEFCom2014 dataset, attributed to the self-attention mechanism's capture of global dependencies in long historical windows (96 hours). RNNs often exhibit memory degradation in such long sequences. Notably, on the UCI Electricity dataset with 370 nodes, GCN-LSTM and GraphSAGE outperform Informer, which treats data as univariate,

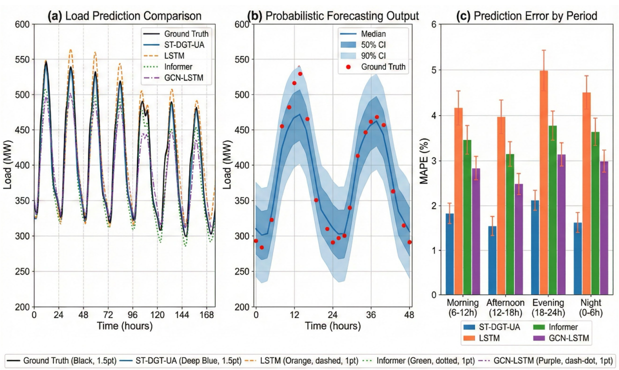

Figure 2. Prediction performance comparison on a representative test week from the UCI Electricity dataset

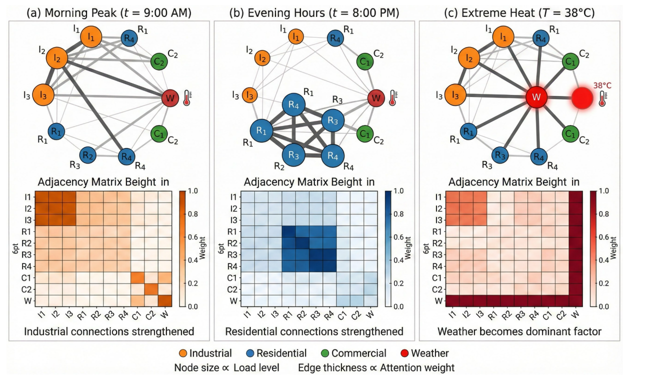

Figure 3. Visualization of learned dynamic adjacency matrices at different time periods

Table 2. Probabilistic forecasting performance comparison

Model	Pinball Loss	CRPS	Winkler
Quantile LSTM	0.045	0.082	45.2
Quantile Informer	0.038	0.065	38.6
DeepAR	0.035	0.061	35.4
ST-DGT-UA	**0.024**	**0.043**	**24.8**

indicating that ignoring spatial correlations is a primary source of error in multi-node scenarios. The proposed ST-DGT-UA model combines Transformer's long-sequence advantages with dynamic graph learning. Compared to the static graph model GCN-LSTM, ST-DGT-UA achieves approximately 24% MAPE reduction on the UCI dataset (from 3.50% to 2.65%), strongly demonstrating that dynamic adjacency matrices capture implicit correlations indescribable by static physical topology, thereby improving prediction accuracy.

Fig. 2 illustrates the prediction results of ST-DGT-UA compared with baseline models on a representative test week. The proposed model demonstrates superior tracking capability for both peak and valley loads, with notably smaller prediction errors during rapid load transitions.

5.2. Probabilistic Forecasting Performance

In smart grid dispatching, prediction reliability is more critical than mere precision. Table 2 presents probabilistic distribution quality evaluation using Pinball Loss and CRPS.

ST-DGT-UA achieves optimal performance across all probabilistic metrics. DeepAR [33], as a classical probabilistic forecasting model, outperforms simple Quantile LSTM, but its CRPS (0.061) remains significantly higher than ST-DGT-UA (0.043). This is primarily because DeepAR's RNN architecture struggles with complex spatio-temporal correlations. The Winkler Score reflects prediction interval width and coverage. ST-DGT-UA's lowest score indicates that its generated confidence intervals are narrower (sharper) while accurately covering true values. This means dispatchers can make decisions based on more precise intervals, reducing unnecessary reserve capacity costs.

5.3. Dynamic Graph Structure Visualization

To verify whether the dynamic graph learning module genuinely learns meaningful spatial relationships, we extracted and visualized adjacency matrix weights generated by the model at different time points, as shown in Fig. 3. During daytime working hours (9:00–17:00), connection weights between commercial zone nodes increase significantly, while evening hours show strengthened weights between residential area nodes, highly consistent with human activity patterns. During simulated extreme high-temperature days, connection weights between all nodes and meteorological nodes universally increase, indicating that the model automatically identifies temperature as the dominant factor at the current moment. This interpretability is absent in traditional black-box deep learning models and holds significant importance for grid dispatchers understanding model decision logic.

5.4. Ablation Study

To quantify the contribution of each core module, we conducted detailed ablation experiments. The variant without Dynamic Graph removes dynamic graphs and uses only the Transformer for independent prediction. The variant without Time2Vec uses traditional one-hot temporal encoding. The variant without ProbSparse employs a standard full attention mechanism. Results are presented in Table 3.

Removing dynamic graphs causes MAPE to deteriorate substantially (+0.8%), proving that spatial information is indispensable in multi-node prediction. Time2Vec removal increases error, indicating that continuous temporal periodic features are critical for capturing daily and weekly patterns. Using full attention

Table 3. Ablation study results on UCI Electricity dataset

Variant	MAPE	RMSE	Params	Time
ST-DGT-UA (Full)	**2.65**	**42.3**	1.25M	120s
w/o Dynamic Graph	3.45	55.2	0.95M	105s
w/o Time2Vec	2.98	48.6	1.25M	120s
w/o ProbSparse	2.62	42.0	1.25M	450s

(Full Attention) yields marginal precision improvement (2.62% vs 2.65%) but increases training time nearly threefold (450s vs 120s). ProbSparse maintains precision while significantly reducing computational cost, enabling the model to process large-scale data with limited computational resources.

6. Discussion

The ST-DGT-UA model proposed in this paper not only performs excellently on academic metrics but also possesses enormous practical application potential in actual power grid operations. High-precision point prediction (MAPE $<$ 2%) can help dispatchers formulate more compact Unit Commitment plans, reducing unnecessary spinning reserve capacity and thereby directly lowering coal consumption and carbon emissions. Confidence intervals output by probabilistic forecasting provide a quantified basis for risk assessment. For instance, when the 95% quantile predicted load exceeds transformer rated capacity, even if point prediction does not exceed limits, dispatchers should trigger warnings and initiate demand response programs, eliminating potential incidents before they develop.

The dynamic graph visualization functionality can assist maintenance personnel in identifying abnormal correlation patterns in the power grid, aiding in determining potential fault propagation paths. This interpretability is crucial for real-world deployment, where operators need to understand and trust model decisions before acting on them.

Despite its advantages, the model has limitations. For cold-start scenarios, such as newly built communities or newly installed smart meters, the dynamic graph learning module struggles to initialize accurate adjacency weights due to a lack of historical data. Future work can explore meta-learning methods [34] to leverage prior knowledge from other similar nodes for rapid adaptation. Regarding data privacy, this paper assumes centralized data storage, but in practice, data from different regional power grids may be restricted from sharing due to privacy regulations. Future work should integrate federated learning techniques [35] to enable collaborative model training while protecting data privacy.

Concerning computational resources, although ProbSparse Attention is employed, dynamic graph construction still involves $O(N^2)$ computation. For ultra-large-scale power grids with over 10,000 nodes, sampling-based graph learning algorithms require further exploration to reduce complexity.

7. Conclusion

This paper presents ST-DGT-UA, an end-to-end short-term load forecasting framework that combines dynamic graph learning, efficient temporal attention, and quantile regression for uncertainty estimation. By learning time-varying adjacency matrices from node representations and using ProbSparse attention for long sequences, the model captures evolving cross-node dependencies while keeping computation manageable.

On GEFCom2014 and UCI Electricity Load Diagrams, ST-DGT-UA achieves MAPE/RMSE of 1.92/145.6 and 2.65/42.3, respectively, and improves probabilistic quality with Pinball Loss 0.024 and CRPS 0.043. Ablation results further indicate that dynamic graph learning substantially improves multi-node forecasting accuracy, while ProbSparse attention preserves accuracy and reduces training time compared with full attention.

Future work will extend the framework to privacy-preserving training across regions (e.g., federated learning) and explore sparsification/sampling strategies for scaling dynamic graph construction to ultra-large power systems; we also plan to incorporate additional exogenous signals such as weather alerts and event indicators for improved robustness under extreme conditions.

References

[1] T. Cao, Y. Xu, G. Liu, S. Tao, W. Tang, and H. Sun, “Feature-enhanced deep learning method for electric vehicle charging demand probabilistic forecasting of charging station,” *Appl. Energy*, vol.371, 123751, 2024.

[2] Y. Eren and I. Kucukdemiral, “A comprehensive review on deep learning approaches for short-term load forecasting,” *Renewable Sustainable Energy Rev.*, vol.189, 114031, 2024.

[3] W. Kong, Z.Y. Dong, Y. Jia, D.J. Hill, Y. Xu, and Y. Zhang, “Short-term residential load forecasting based on LSTM recurrent neural network,” *IEEE Trans. Smart Grid*, vol.10, no.1, pp.841–851, 2019.

[4] K. Chen, K. Chen, Q. Wang, Z. He, J. Hu, and J. He, “Short-term load forecasting with deep residual networks,” *IEEE Trans. Smart Grid*, vol.10, no.4, pp.3943–3952, 2019.

[5] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long, “iTransformer: Inverted Transformers are effective for time series forecasting,” *Proc. Int. Conf. Learning Representations*, 2024.

[6] H. Jiang and W. Zheng, "Deep learning with regularized robust long- and short-term memory network for probabilistic short-term load forecasting," *J. Forecasting*, vol.41, no.6, pp.1201–1216, 2022.
[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol.9, no.8, pp.1735–1780, 1997.
[8] J. Wang, K. Wang, Z. Li, H. Lu, H. Jiang, and Q. Xing, "A multitask integrated deep-learning probabilistic prediction for load forecasting," *IEEE Trans. Power Syst.*, vol.39, no.1, pp.1240–1250, 2024.
[9] W.L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *Proc. Advances in Neural Information Processing Systems*, vol.30, pp.1024–1034, 2017.
[10] Y. Wang, L. Rui, J. Ma, and Q. Jin, "A short-term residential load forecasting scheme based on the multiple correlation-temporal graph neural networks," *Appl. Soft Comput.*, vol.146, 110629, 2023.
[11] C. Wei, D. Pi, M. Ping, and H. Zhang, "Short-term load forecasting using spatial-temporal embedding graph neural network," *Electric Power Systems Research*, vol.225, 109873, 2023.
[12] L. Chen, D. Chen, Z. Shang, B. Wu, C. Zheng, B. Wen, and W. Zhang, "Multi-scale adaptive graph neural network for multivariate time series forecasting," *IEEE Trans. Knowl. Data Eng.*, vol.35, no.10, pp.10748–10761, 2023.
[13] G. Jin, Y. Liang, Z. Fang, Z. Shao, J. Huang, J. Zhang, and Y. Zheng, "Spatio-temporal graph neural networks for predictive learning in urban computing: A survey," *IEEE Trans. Knowl. Data Eng.*, vol.36, no.10, pp.5388–5408, 2024.
[14] J. Liu, S. Dong, P. Zhang, T. Li, C. Peng, and Z. Hu, "Load forecasting based on dynamic adaptive and adversarial graph convolutional networks," *Energy Build.*, vol.312, 114206, 2024.
[15] H. Jiang, Y. Dong, Y. Dong, and J. Wang, "Power load forecasting based on spatial–temporal fusion graph convolution network," *Technological Forecasting and Social Change*, vol.204, 123435, 2024.
[16] Y. Lv, L. Wang, D. Long, Q. Hu, and Z. Hu, "Multi-area short-term load forecasting based on spatiotemporal graph neural network," *Engineering Applications of Artificial Intelligence*, vol.138, 109398, 2024.
[17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *Proc. Advances in Neural Information Processing Systems*, vol.30, pp.5998–6008, 2017.
[18] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "TimesNet: Temporal 2D-variation modeling for general time series analysis," *Proc. Int. Conf. Learning Representations*, 2023.
[19] Y. Nie, N.H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with Transformers," *Proc. Int. Conf. Learning Representations*, 2023.
[20] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, and R. Jin, "FEDformer: Frequency enhanced decomposed Transformer for long-term series forecasting," *Proc. Int. Conf. Machine Learning*, vol.162, pp.27268–27286, 2022.
[21] J.W. Chan and C.K. Yeo, "A Transformer based approach to electricity load forecasting," *The Electricity Journal*, vol.37, no.2, 107370, 2024.
[22] H. Jiang, S. Pan, Y. Dong, and J. Wang, "Probabilistic electricity price forecasting based on penalized temporal fusion transformer," *J. Forecasting*, vol.43, no.5, pp.1465–1491, 2024.
[23] B. Wang, M. Mazhari, and C.Y. Chung, "A novel hybrid method for short-term probabilistic load forecasting in distribution networks," *IEEE Trans. Smart Grid*, vol.13, no.5, pp.3650–3661, 2022.
[24] J. Gawlikowski, C.R.N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher, M. Shahzad, W. Yang, R. Bamler, and X.X. Zhu, "A survey of uncertainty in deep neural networks," *Artif. Intell. Rev.*, vol.56, pp.1513–1589, 2023.
[25] X. Tang, H. Chen, W. Xiang, J. Yang, and M. Zou, "Short-term load forecasting using channel and temporal attention based temporal convolutional network," *Electric Power Systems Research*, vol.205, 107761, 2022.
[26] Q. Wen, T. Zhou, C. Zhang, W. Chen, Z. Ma, J. Yan, and L. Sun, "Transformers in time series: A survey," *Proc. Int. Joint Conf. Artificial Intelligence*, pp.6778–6786, 2023.
[27] T. Hong, P. Pinson, S. Fan, H. Zareipour, A. Troccoli, and R.J. Hyndman, "Probabilistic energy forecasting: Global Energy Forecasting Competition 2014 and beyond," *Int. J. Forecasting*, vol.32, no.3, pp.896–913, 2016.
[28] A. Trindade, "ElectricityLoadDiagrams20112014 Dataset," UCI Machine Learning Repository, 2015.
[29] B.N. Oreshkin, D. Carpov, N. Chapados, and Y. Bengio, "N-BEATS: Neural basis expansion analysis for interpretable time series forecasting," *Proc. Int. Conf. Learning Representations*, 2020.
[30] T.N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *Proc. Int. Conf. Learning Representations*, 2017.
[31] T. Gneiting and A.E. Raftery, "Strictly proper scoring rules, prediction, and estimation," *J. Am. Stat. Assoc.*, vol.102, no.477, pp.359–378, 2007.
[32] F.X. Diebold and R.S. Mariano, "Comparing predictive accuracy," *J. Bus. Econ. Stat.*, vol.13, no.3, pp.253–263, 1995.
[33] D. Salinas, V. Flunkert, J. Gasthaus, and T. Januschowski, "DeepAR: Probabilistic forecasting with autoregressive recurrent networks," *Int. J. Forecasting*, vol.36, no.3, pp.1181–1191, 2020.
[34] K. Li, Z. Li, C. Huang, and Q. Ai, "Online transfer learning-based residential demand response potential forecasting for load aggregator," *Appl. Energy*, vol.358, 122631, 2024.
[35] Z. Chen, J. Li, L. Cheng, and X. Liu, "Federated-WDCGAN: A federated smart meter data sharing framework for privacy preservation," *Appl. Energy*, vol.334, 120711, 2023.