Dynamic Graph-Augmented Transformers for Exogenous-Aware Time Series Forecasting

1st Tianxiao Ren
Zhejiang University
Hangzhou, China
22360017@zju.edu.cn

2nd Zhen Wang
Zhejiang University
Hangzhou, China
12460103@zju.edu.cn

3rd Yunzhi Hao
Zhejiang University
Hangzhou, China
hyzhi@hzcu.edu.cn

4th Yang Gao
Zhejiang University
Hangzhou, China
roygao@zju.edu.cn

5th Shunyu Liu
Zhejiang University
Hangzhou, China
liushunyu@zju.edu.cn

6th Kaixuan Chen*
Zhejiang University
Hangzhou, China
chenkx@zju.edu.cn

7th Mingli Song
Zhejiang University
Hangzhou, China
brooksong@zju.edu.cn

***Abstract*—Exogenous-aware time series forecasting requires capturing complex temporal dependencies within endogenous signals and incorporating heterogeneous exogenous variates that influence the target variables. However, existing approaches often use static attention based on a single similarity computation, which makes it hard to capture temporal dependencies, and use simple encodings for exogenous inputs that may miss important covariate patterns. Therefore, we propose Dynamic Graph-Augmented Transformers (DGAFormer), which equips Transformer attention with our Dynamic Graph Attention (DGA) mechanism and incorporates a Gated Multi-scale Exogenous (GME) encoder for covariate representation. Specifically, our DGA mechanism augments attention with dynamic graph-based refinement, alleviating the limitation of static attention. Using this mechanism, DGAFormer applies DGA-based self-attention to learn endogenous dependencies and DGA-based cross-attention, where a global endogenous token queries exogenous tokens, to incorporate exogenous information consistently. In addition, to provide more informative covariate representations, we design a GME encoder that extracts patterns at multiple temporal resolutions using a convolutional bank and adaptively fuses them into compact exogenous tokens. Extensive experiments on multiple real-world benchmarks show that DGAFormer consistently outperforms strong baselines, with particularly pronounced gains in covariate-rich settings.**

***Index Terms*—Time series forecasting, Dynamic graph attention, Graph neural networks, Multi-scale feature extraction**

I. Introduction

Time series forecasting is a fundamental problem in machine learning and is widely used in energy load forecasting, traffic flow prediction, and economic indicator analysis [1]–[4]. In many real-world applications, the target series is driven not only by its historical dynamics but also by external factors. Importantly, many of these external drivers are observable as time-varying exogenous variates and can be available at prediction time. For example, weather conditions affect electricity demand, while holidays and promotional events can shift consumer behavior and change sales patterns. Therefore, exogenous-aware forecasting, which explicitly incorporates such exogenous variates to better predict future values of the endogenous target, has become an important direction for improving forecasting accuracy [5], [6].

Recent deep learning methods incorporate exogenous variables by first transforming them into latent representations and then combining them with the target series to improve forecasting performance. Many approaches follow an encoder-and-fusion design, where exogenous inputs are embedded and integrated with endogenous representations either at the input stage or within intermediate layers [7]. This paradigm allows the model to jointly leverage endogenous dynamics and auxiliary covariate signals within a unified forecasting pipeline. Meanwhile, a large body of recent forecasters adopts attention-based architectures to model temporal dependencies, using attention to capture relationships across time steps [8], [9]. Overall, these designs offer a practical way to use both endogenous history and exogenous variates and have become common in modern time series forecasting.

Despite recent progress, two gaps remain in exogenous-aware forecasting. First, many methods rely on static, one-step attention to model temporal dependencies, where interactions are computed from a single round of pairwise similarity and remain fixed within a layer. However, dependencies in real-world time series are often context-dependent and unfold through intermediate steps, so the influence of an earlier time point may propagate before it manifests in future values. Such multi-step propagation is difficult to represent with one-step attention, which leads to under-modeling of higher-order dependencies and reduced accuracy under complex dynamics. Second, exogenous variables are frequently encoded in a coarse manner, e.g., via simple projections or pooling over the entire input window, which emphasizes global statistics and weakens the temporal localization of covariate effects. In practice, exogenous effects are often time-local and stage-dependent, and summarizing them into a global representation may fail to preserve the temporal localization of covariate impacts, particularly near abrupt changes.

Motivated by these observations, we propose **DGAFormer**, which tackles multi-step dependency modeling and tempo-

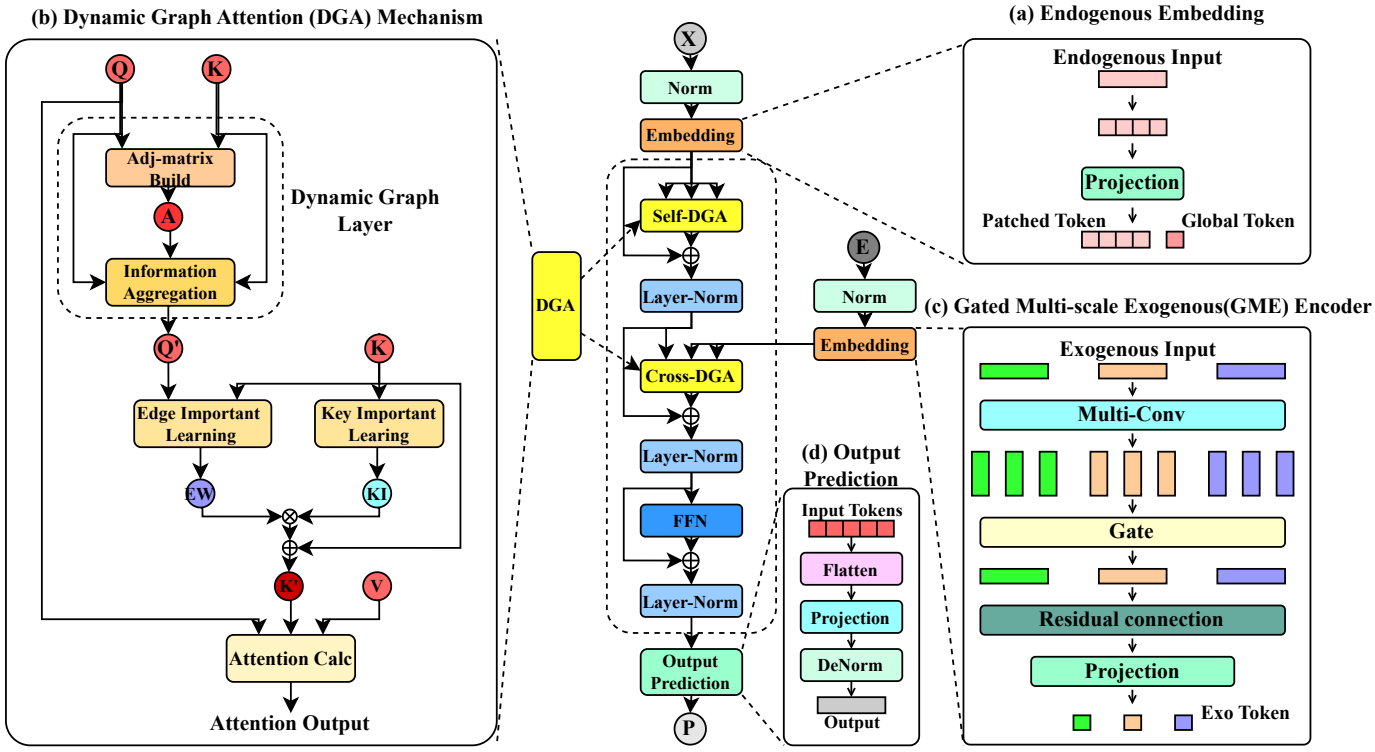

Fig. 1. The Structure of DGAFormer. (a) The endogenous embedding module forms patch tokens and a learnable global token from the historical target series. (b) Dynamic Graph Attention (DGA) enhances attention with a context-dependent dynamic graph update to capture multi-hop temporal dependencies. (c) The Gated Multi-scale Exogenous (GME) extracts multi-scale patterns from each exogenous variable and produces exogenous tokens. (d) Output prediction.

rally adaptive exogenous integration in a unified forecasting framework. Our design is motivated by the fact that real-world dependencies may propagate through intermediate steps beyond one-step interactions, while exogenous effects are often localized and multi-scale, which is not well captured by static attention and global exogenous variate summarization. Accordingly, DGAFormer introduces a **Dynamic Graph Attention (DGA)** mechanism that augments attention with a context-dependent graph refinement, so temporal tokens can exchange information beyond direct one-step interactions and obtain richer multi-hop context. This mechanism is used in two places in a unified way: *endogenous self-attention* focuses on learning temporal structures within endogenous tokens, while *exogenous cross-attention* uses a global endogenous token to query exogenous tokens and inject exogenous information into the endogenous representation. In this design, endogenous modeling and exogenous injection share the same DGA mechanism, which keeps the architecture consistent and avoids treating exogenous variates as an ad-hoc add-on. To further ensure that exogenous variates are represented in a way that matches how they influence the target, we also design a **Gated Multi-scale Exogenous (GME)** encoder. Instead of compressing exogenous inputs into a single global feature, it extracts patterns at multiple temporal resolutions and adaptively fuses them, so local fluctuations and global trends can both be preserved when they are informative. Together, DGA and GME form a unified exogenous-aware framework that improves forecasting accuracy by strengthening multi-hop dependency modeling and multi-scale exogenous variates representation.

In summary, our contributions are as follows:

- We propose DGAFormer, an exogenous-aware forecasting framework that jointly models temporal dependencies and incorporates exogenous information in a unified architecture.
- We introduce Dynamic Graph Attention (DGA) and instantiate it in both endogenous self-attention and exogenous cross-attention, enabling context-adaptive modeling of multi-hop dependencies and consistent exogenous information injection.
- We design a Gated Multi-scale Exogenous (GME) encoder that extracts exogenous patterns at multiple temporal resolutions and adaptively fuses them into compact and informative exogenous representations.
- Extensive experiments on multiple real-world benchmarks demonstrate that DGAFormer consistently outperforms strong baselines, with more pronounced gains on exogenous-rich datasets.

II. Relation Work

A. Exogenous Variables in Time Series Forecasting

Exogenous variables have long been studied in time series forecasting. Classical statistical models such as ARIMAX [10] and SARIMAX [11] incorporate exogenous variates through explicit linear terms, which are effective for interpretable linear dependencies but are limited in expressing nonlinear and time-varying influences. With the rise of deep learning, recurrent models and their variants extend forecasting to the nonlinear

regime by injecting exogenous features into the input sequence or fusing them with hidden states through gating mechanisms, enabling flexible interactions between exogenous variates and targets.

More recently, attention-based models [12]–[16] provide a stronger foundation for leveraging exogenous information. For example, TFT [17] employs variable selection networks to identify important exogenous variates, and cross-attention has been used to align target sequences with exogenous sequences [18]. Despite these advances, many methods emphasize feature selection or global fusion and still treat the temporal structure of exogenous series in a coarse manner, often relying on simple embeddings or window-level summaries. However, exogenous effects in practice can be highly local and manifest at different temporal scales. In contrast, our work explicitly models exogenous variates by extracting multi-scale local patterns and adapting their contributions, aiming to provide finer-grained and more informative exogenous representations.

B. Temporal Dependency Modeling Mechanisms

Modeling temporal dependencies is central to forecasting. Early approaches primarily relied on recurrent neural networks (RNNs) and their variants to capture sequential patterns via recurrent transitions [19], yet they may struggle with long-range dependencies due to gradient issues. Temporal Convolutional Networks (TCNs) expand the receptive field by dilated convolutions [20], but their dependency structure is still governed by fixed convolution kernels, which limits context-adaptive interaction across time steps.

Transformers [12] address this limitation by using self-attention to model dependencies among all time steps and have become a dominant framework for time series forecasting. Many studies improve attention by injecting priors such as trend and seasonality [9], [21], augmenting the architecture with recurrence or memory [22], [23], or accelerating attention computation via kernelization and low-rank approximations [24], [25]. Nevertheless, in standard attention, Keys are produced by fixed projections, which makes it difficult to reflect context-dependent and multi-hop temporal associations. In parallel, graph neural networks (GNNs) [26]–[29] have been used for forecasting [30], [31], mainly either by modeling inter-variable relations with a predefined graph (e.g., spatio-temporal graphs) [32], [33] or by replacing attention with message passing over time [34]. These paradigms either overlook dynamic relations between time steps within a sequence or lose the efficient global interaction offered by attention, while graph-structured Transformer variants often depend on fixed graphs [35]. Our work bridges these lines by embedding a dynamic graph aggregation module into attention to generate context-aware Keys, thereby capturing multi-hop temporal structure while preserving the ability and global modeling benefits of dot-product attention.

III. Methodology

In exogenous-aware time series forecasting, we are given a multivariate endogenous time series and a set of exogenous variates. Specifically, let $X^{\mathrm{endo}} \in \mathbb{R}^{N\times T}$ denote the historical endogenous variables over T time steps, where N is the number of target channels. Let $X^{\mathrm{exo}} \in \mathbb{R}^{D\times T}$ be the historical exogenous variables, and $Y^{\mathrm{exo}} \in \mathbb{R}^{D\times F}$ be the future exogenous variables over the forecasting horizon F (if available). The goal is to predict the future endogenous values $Y^{\mathrm{endo}} \in \mathbb{R}^{N\times F}$ by learning a forecaster $f_\theta(\cdot)$:

$$\widehat{Y}^{\mathrm{endo}} = f_\theta\big(X^{\mathrm{endo}},\, X^{\mathrm{exo}},\, Y^{\mathrm{exo}}\big)\,. \tag{1}$$

When future exogenous variables are unavailable, we set $Y^{\mathrm{exo}} = \varnothing$ and the model reduces to forecasting with historical endogenous and exogenous inputs only.

Figure 1 presents the architecture of **DGAFormer** with four components: **(a) Endogenous Embedding**, which encodes the historical target series into token representations; **(b) Dynamic Graph Attention Mechanism**, which enhances dependency modeling by introducing a context-dependent dynamic graph update into attention; **(c) Gated Multi-scale Exogenous Encoder**, which extracts exogenous patterns at multiple temporal resolutions and adaptively fuses them into informative representations; and **(d) Output prediction**, which maps the final representation to the forecasting outputs with an optional de-normalization step.

A. Endogenous Embedding

Given the historical endogenous series $X^{\mathrm{endo}} \in \mathbb{R}^{N\times T}$, we first apply instance-wise normalization along the temporal dimension by the normalization operator $\mathrm{Norm}(\cdot)$:

$$\widetilde{X}^{\mathrm{endo}} = \mathrm{Norm}(X^{\mathrm{endo}}; \mu, \sigma) = (X^{\mathrm{endo}} - \mu)/\sigma, \tag{2}$$

where $\mu \in \mathbb{R}^{N\times 1}$ and $\sigma \in \mathbb{R}^{N\times 1}$ are computed from the input window.

We then tokenize $\widetilde{X}^{\mathrm{endo}}$ into patch tokens and a learnable global token. Let the patch length be p and the number of patches be $L_p = \lfloor T/p \rfloor$. For the i-th patch, we take a slice $\widetilde{X}^{\mathrm{endo}}_{[:,(i-1)p+1:ip]} \in \mathbb{R}^{N\times p}$, flatten it into a vector in $\mathbb{R}^{Np}$, and project it to the model dimension d:

$$z_i^{\mathrm{endo}} = \mathrm{ValEmb}\Big(\mathrm{vec}\Big(\widetilde{X}^{\mathrm{endo}}_{[:,(i-1)p+1:ip]}\Big)\Big) + \mathrm{PosEmb}(i). \tag{3}$$

This design encodes the target channel dimension N into each token content (via $\mathrm{vec}(\cdot)$), while the token length is determined by the temporal segmentation. We append a learnable global token $z^{\mathrm{endo}}_{\mathrm{glb}} \in \mathbb{R}^d$ following [7] to obtain

$$Z^{\mathrm{endo}} = \Big[z_1^{\mathrm{endo}},\, \ldots,\, z_{L_p}^{\mathrm{endo}},\, z_{\mathrm{glb}}^{\mathrm{endo}}\Big] \in \mathbb{R}^{(L_p+1)\times d}. \tag{4}$$

B. Dynamic Graph Attention Mechanism

We propose Dynamic Graph Attention (DGA) to enhance attention keys by introducing context-dependent, multi-hop interactions. Consider a generic attention block with

$$Q \in \mathbb{R}^{n_q\times d}, \qquad K \in \mathbb{R}^{n_k\times d}, \qquad V \in \mathbb{R}^{n_k\times d}, \tag{5}$$

where n_q and n_k are the numbers of query tokens and key/value tokens, respectively. Standard scaled dot-product attention is

$$\mathrm{Attn}(Q,K,V) = \mathrm{Softmax}\left(\frac{QK^\top}{\sqrt{d}}\right)V. \tag{6}$$

We embed a *dynamic graph layer* into attention by (i) propagating multi-hop information to obtain an updated query Q', (ii) learning edge-wise and key-wise importance to enhance keys into K', and (iii) computing attention using (Q, K', V).

i) Dynamic graph propagation. DGA constructs a context-dependent affinity matrix and iteratively propagates information: initialize $Q_0 = Q$ and define

$$A_{j-1} = Q_{j-1}K^\top \in \mathbb{R}^{n_q \times n_k}, \tag{7}$$

$$H_j = A_{j-1}K \in \mathbb{R}^{n_q \times d}, \tag{8}$$

$$Q_j = \mathrm{Conv}(Q_{j-1} + H_j) \in \mathbb{R}^{n_q \times d}, \tag{9}$$

where $\mathrm{Conv}(\cdot)$ denotes a lightweight convolution applied along the token dimension, $j = 1, 2, \ldots, J$ and J is the propagation depth.

ii) Edge-wise and key-wise importance. We compute an edge-wise importance matrix $EW \in \mathbb{R}^{n_q \times n_k}$ by a pairwise scorer:

$$EW_{a,b} = \mathrm{Sigmoid}(\phi([q_{J,a};\, k_b])), \tag{10}$$

where $q_{J,a}$ is the a-th row of Q_J, k_b is the b-th row of K, $[\cdot;\cdot]$ denotes concatenation, $\phi(\cdot)$ is an MLP. We also compute a key-wise importance vector $KI \in \mathbb{R}^{n_k}$:

$$KI_b = \mathrm{Sigmoid}(\psi(k_b)), \tag{11}$$

where $\psi(\cdot)$ is another MLP. Then KI is expanded as $\tilde{KI} = \mathrm{Expand}(KI) \in \mathbb{R}^{1 \times n_k}$. Combining them gives the final edge weights $W \in \mathbb{R}^{n_q \times n_k}$:

$$W_{a,b} = EW_{a,b} \odot \tilde{KI}_b, \tag{12}$$

where $\odot$ denotes element-wise multiplication.

iii) Key enhancement. We enhance keys by aggregating query-dependent messages back to the key side:

$$\Delta K = \mathrm{Proj}(W) \in \mathbb{R}^{n_q \times n_k \times d}, \tag{13}$$

$$K' = \mathrm{Expand}(K) + \alpha\, \Delta K \in \mathbb{R}^{n_q \times n_k \times d}, \tag{14}$$

where $\mathrm{Proj}(\cdot)$ is a linear projection and α controls the enhancement strength. Similarly, Q also needs to expand its dimensions to $\mathbb{R}^{n_q \times 1 \times d}$.

Finally, DGA replaces K with K' in attention:

$$\mathrm{DGA}(Q,K,V) = \mathrm{Softmax}\left(\frac{\mathrm{Expand}(Q){K'}^\top}{\sqrt{d}}\right)V. \tag{15}$$

In DGAFormer, DGA is a generic mechanism that operates on token embeddings, and it can be used for endogenous self-attention as well as for endogenous-to-exogenous cross-attention. *(i) Self-DGA for endogenous modeling:* given the endogenous tokens Z^{endo} (Section III-A), we apply DGA in a self-attention manner to capture temporal dependencies within the endogenous sequence. *(ii) Cross-DGA for exogenous injection:* to incorporate exogenous information, we first obtain exogenous tokens Z^{exo} from the proposed Gated Multi-scale Exogenous (GME) encoder (Section III-C), and then apply DGA as a cross-attention module where the global endogenous token attends to Z^{exo}.

C. Gated Multi-scale Exogenous Encoder

Exogenous variables may exert localized and scale-varying effects. We therefore introduce a gated multi-scale encoder to produce compact exogenous tokens.

i) Multi-scale temporal features. For the i-th exogenous variable $x_i^{\mathrm{exo}} \in \mathbb{R}^T$, we first apply a bank of 1D convolutions with kernel sizes $\{k_s\}_{s=1}^S$ to capture multi-scale patterns:

$$F_{i,s} = \mathrm{Conv}_{k_s}(x_i^{\mathrm{exo}}) \in \mathbb{R}^T. \tag{16}$$

A simple kernel schedule is $k_s = 2^s + 1$ with an upper bound $k_s \le T/4$ to limit excessive padding.

ii) Gated fusion across scales. First, we compute scale gates

$$g_{i,s} = \mathrm{Sigmoid}(\xi(F_{i,s})) \in \mathbb{R}^T, \tag{17}$$

where $\xi(\cdot)$ is an MLP. Then we fuse multi-scale features by

$$F_i = \sum_{s=1}^{S} g_{i,s} \odot F_{i,s} \in \mathbb{R}^T. \tag{18}$$

We form a residual-enhanced feature map $\widetilde{x}_i = x_i^{\mathrm{exo}} + F_i \in \mathbb{R}^T$ and obtain one exogenous token per variable via temporal pooling:

$$z_i^{\mathrm{exo}} = \widetilde{x}_i W_i + b_i \in \mathbb{R}^d, \tag{19}$$

where $W_i \in \mathbb{R}^{T \times d}$ and $b_i \in \mathbb{R}^d$.

Stacking all tokens yields

$$Z^{\mathrm{exo}} = [z_1^{\mathrm{exo}}, \ldots, z_D^{\mathrm{exo}}]^\top \in \mathbb{R}^{D \times d}, \tag{20}$$

which is fed into Cross-DGA (Section III-B), where the global endogenous token serves as the query and the exogenous tokens act as key/value, so that exogenous information is injected into the endogenous representation.

D. Output Prediction

After L DGAFormer encoder layers, we obtain the final endogenous tokens $Z^{\mathrm{endo},(L)} \in \mathbb{R}^{(L_p+1) \times d}$. and apply a prediction head to produce an intermediate forecast:

$$\widehat{Y}_{\mathrm{pred}}^{\mathrm{endo}} = \mathrm{Flatten}(Z^{\mathrm{endo},(L)})W_o + b_o \in \mathbb{R}^F, \tag{21}$$

where $\mathrm{Flatten}(\cdot)$ donates flattening, $W_o \in \mathbb{R}^{((L_p+1)\times d) \times F}$ and $b_o \in \mathbb{R}^F$.

Since the endogenous input is normalized in Eq. (2), we output the final prediction in the original scale via a denormalization operator $\mathrm{DeNorm}(\cdot)$:

$$\widehat{Y}^{\mathrm{endo}} = \mathrm{DeNorm}\Big(\widehat{Y}_{\mathrm{pred}}^{\mathrm{endo}}; \mu, \sigma\Big) = \widehat{Y}_{\mathrm{pred}}^{\mathrm{endo}} \odot \sigma + \mu, \tag{22}$$

where μ, σ are the same instance-wise statistics used in Eq. (2) and are broadcast along the horizon dimension.

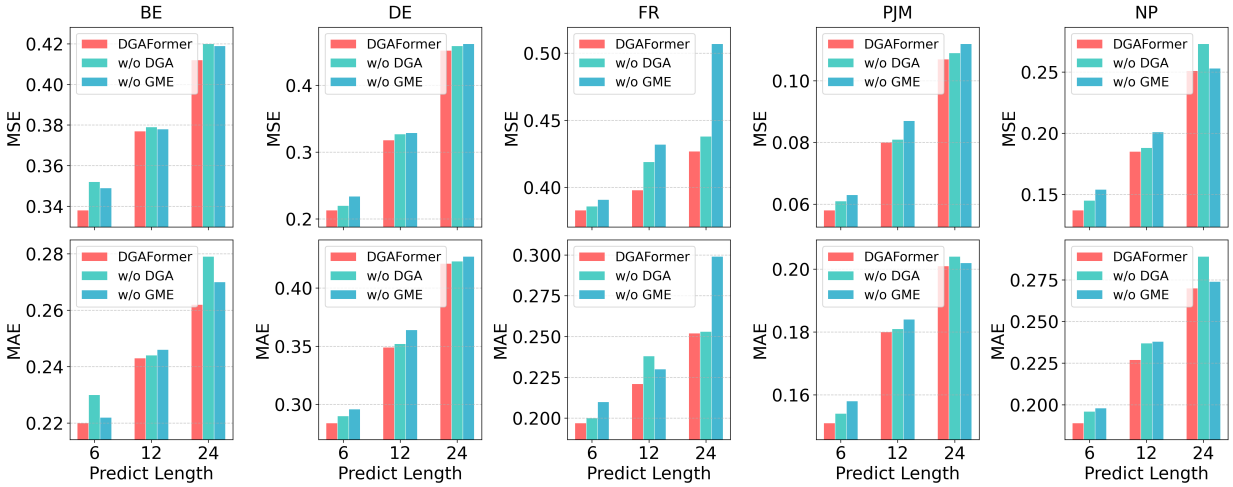

Fig. 2. The Results of Ablation Study. w/o DGA means replacing DGA with standard dot-product attention, and w/o GME means replacing the GME with a single linear projection, with all other settings identical.

IV. Experiments

We evaluate DGAFormer on multiple real-world benchmarks for exogenous-aware time series forecasting. All datasets include endogenous targets and exogenous variates, with varying exogenous variate dimensionalities across domains.

A. Experiment sets

We evaluate on six public multivariate forecasting benchmarks spanning electricity markets, weather, and energy systems, where each dataset contains an endogenous target and exogenous variates. **EPF** [36] includes five regional subsets (**NP**, **PJM**, **BE**, **FR**, **DE**); the target is electricity price with two market-related exogenous variates. **Weather** [8] contains 21 meteorological variables (10-minute); we predict *Wet Bulb* using the remaining variables as exogenous variates. **ETT** [8] consists of ETTh1/ETTh2 (hourly) and ETTm1/ETTm2 (15-minute); the target is oil temperature with six load exogenous variates. We compare against strong baselines, including Transformer-based (TimeXer, iTransformer), MLP-based (TiDE), and CNN-based (SCINet [37]) models. Dataset statistics are summarized in Table I.

TABLE I
Details of Datasets

Datasets	Ex. Number	Sampling Freq.	Dataset Size
BE	2	1 Hour	(36500, 5219, 10460)
DE	2	1 Hour	(36500, 5219, 10460)
FR	2	1 Hour	(36500, 5219, 10460)
NP	2	1 Hour	(36500, 5219, 10460)
PJM	2	1 Hour	(36500, 5219, 10460)
ETTh	6	1 Hour	(8545, 2881, 2881)
ETTm	6	15 Minutes	(34465, 11521, 11521)
Weather	20	10 Minutes	(36792, 5271, 10540)

EPF and Weather are split into training/validation/test sets with a 7:1:2 ratio while ETT are split with 3:1:1. Unless otherwise specified, the input length is set to 96 and the forecasting horizons are {6,12,24}.

B. Main Results

Tables II–III summarize the performance of DGAFormer and strong baselines on EPF, ETT, and Weather, where lower MAE and MSE indicate better accuracy. The MSE values of Weather in Table III are the coefficients, with an order of magnitude of 10^{-4}. The best and second-best results are highlighted in **bold** and underline, respectively.

i) Results on EPF. Table II shows that DGAFormer achieves the best average performance across all horizons and metrics, and attains the majority of the best results across the five EPF subsets. Although DGAFormer is occasionally ranked second on certain EPF subsets, the overall trend indicates consistent competitiveness across markets.

ii) Results on ETT and Weather. As shown in Tables III, DGAFormer delivers uniformly strong performance on both ETT and Weather, achieving the best results in nearly all comparisons and the best average metrics across horizons. These gains are particularly pronounced on Weather, which contains substantially more exogenous variates than EPF.

iii) Discussion. Across datasets, DGAFormer exhibits the most consistent overall accuracy, while the margin to the strongest baselines varies with the complexity of exogenous inputs. On EPF, the number of exogenous variates is limited and their temporal patterns are relatively regular, so the benefit of more expressive exogenous modeling is less pronounced and competitive Transformer baselines (e.g., TimeXer) can match or slightly exceed DGAFormer on a few cases. In contrast, on datasets with richer and higher-dimensional exogenous variates (ETT and especially Weather), DGAFormer shows clear advantages, suggesting that dynamic dependency modeling together with improved exogenous representations is particularly effective when exogenous information is diverse and informative.

C. Ablation Study

DGAFormer consists of two key design choices, namely Dynamic Graph Attention (DGA) and the Gated Multi-scale Exogenous (GME) encoder. To assess their individual contributions, we construct two controlled variants by removing

TABLE II
PREDICT RESULTS OF BASELINES AND DGAFORMER ON DATASET EPF

Models	F	BE		DE		FR		PJM		NP		AVG	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
TimeXer	6	0.353	0.226	0.227	0.290	0.417	0.205	**0.058**	0.152	0.149	0.197	0.241	0.214
	12	0.396	0.256	0.330	0.355	0.432	0.224	**0.080**	**0.178**	0.192	0.234	0.286	0.249
	24	0.426	0.275	0.460	0.423	0.467	0.258	**0.106**	0.205	**0.249**	0.273	0.342	0.287
iTransformer	6	0.348	0.233	0.229	0.297	0.424	0.212	0.059	0.154	0.155	0.206	0.243	0.220
	12	0.404	0.270	0.337	0.366	0.438	0.230	**0.080**	**0.178**	0.209	0.249	0.294	0.259
	24	0.436	0.272	0.467	0.433	0.485	0.267	0.107	0.207	0.268	0.292	0.353	0.294
TiDE	6	0.494	0.332	0.393	0.420	0.459	0.251	0.101	0.211	0.266	0.295	0.343	0.302
	12	0.524	0.351	0.509	0.479	0.483	0.279	0.113	0.224	0.305	0.322	0.387	0.331
	24	0.527	0.340	0.573	0.498	0.507	0.292	0.126	0.231	0.338	0.343	0.414	0.341
SCINet	6	0.603	0.385	0.313	0.368	0.625	0.324	0.102	0.214	0.239	0.276	0.376	0.313
	12	0.627	0.400	0.453	0.446	0.642	0.346	0.122	0.234	0.302	0.320	0.429	0.349
	24	0.668	0.423	0.610	0.516	0.675	0.369	0.137	0.245	0.351	0.349	0.488	0.380
DGAFormer	6	**0.338**	**0.220**	**0.213**	**0.284**	**0.383**	**0.197**	**0.058**	**0.151**	**0.137**	**0.189**	**0.226**	**0.208**
	12	**0.377**	**0.243**	**0.318**	**0.349**	**0.398**	**0.221**	**0.080**	0.180	**0.185**	**0.227**	**0.272**	**0.244**
	24	**0.412**	**0.262**	**0.452**	**0.421**	**0.427**	**0.252**	0.107	**0.201**	0.251	**0.270**	**0.330**	**0.281**

TABLE III
PREDICT RESULTS OF BASELINES AND DGAFORMER ON DATASET ETT AND WEATHER

Models	F	ETTh1		ETTh2		ETTm1		ETTm2		Weather		AVG	
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
TimeXer	6	0.018	0.092	0.035	0.124	0.005	0.047	0.004	0.043	2.655	0.011	0.012	0.063
	12	0.030	0.122	0.078	0.186	0.009	0.062	0.010	0.064	4.648	0.015	0.025	0.090
	24	0.046	0.158	0.120	0.247	0.016	0.083	0.027	0.105	7.123	**0.018**	0.042	0.122
iTransformer	6	0.018	0.092	0.041	0.131	0.005	**0.037**	0.003	0.037	3.969	0.013	0.013	0.062
	12	0.030	0.124	0.081	0.191	0.009	0.062	0.010	0.061	4.685	0.015	0.026	0.091
	24	0.048	0.160	0.123	0.250	0.016	0.086	0.029	0.105	7.830	0.020	0.043	0.124
TiDE	6	0.032	0.130	0.101	0.228	0.007	0.055	0.009	0.066	3.313	0.013	0.030	0.098
	12	0.040	0.148	0.125	0.258	0.011	0.071	0.020	0.094	4.484	0.016	0.039	0.117
	24	0.055	0.179	0.152	0.294	0.019	0.095	0.048	0.142	6.645	0.019	0.055	0.146
SCINet	6	0.023	0.110	0.060	0.170	0.006	0.050	0.004	0.040	12.11	0.025	0.019	0.079
	12	0.035	0.136	0.102	0.228	0.010	0.065	0.010	0.064	12.47	0.025	0.032	0.104
	24	0.050	0.168	0.140	0.278	0.017	0.088	0.032	0.111	12.31	0.026	0.048	0.134
DGAFormer	6	**0.010**	**0.076**	**0.023**	**0.104**	**0.003**	0.041	**0.002**	**0.032**	**2.154**	**0.010**	**0.008**	**0.053**
	12	**0.017**	**0.099**	**0.046**	**0.152**	**0.005**	**0.054**	**0.006**	**0.054**	**4.027**	**0.014**	**0.015**	**0.075**
	24	**0.027**	**0.125**	**0.066**	**0.191**	**0.010**	**0.072**	**0.019**	**0.089**	**6.639**	**0.018**	**0.025**	**0.099**

one component at a time while keeping the remaining settings unchanged, i.e., *w/o DGA* and *w/o GME*.

As shown in Fig. 2, discarding DGA consistently increases both MSE and MAE across datasets and forecasting horizons, indicating that the dynamic graph refinement is important for modeling higher-order temporal dependencies that are not well captured by single-pass attention. Similarly, replacing GME with a simpler exogenous encoder also leads to systematic performance drops, which suggests that multi-scale and gated exogenous encoding provides more informative exogenous representations, especially when exogenous effects are localized. Overall, the two components yield complementary benefits: DGA primarily strengthens dependency modeling, whereas GME improves exogenous representation, and their combination delivers the most stable and accurate forecasts.

D. Parameter Sensitivity Study

To identify an appropriate parameter setting, we conduct a parameter sensitivity study to examine the effect of the DGA depth on forecasting performance. Using the DE dataset as the benchmark, we vary the number of DGA layers from 1 to 6 under different prediction horizons. The corresponding test results are shown in Fig. 3.

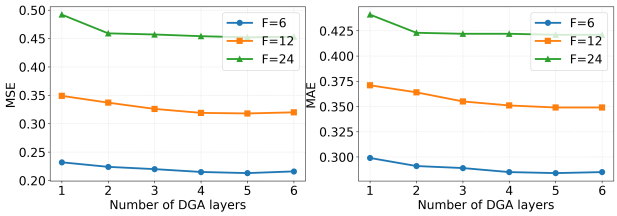

Fig. 3. The Results of parameter sensitivity study.

V. CONCLUSION

In this paper, we studied exogenous-aware time series forecasting, where accurate prediction depends on both effective exogenous modeling and expressive temporal dependency learning. To address the limited ability of prior methods to capture fine-grained exogenous patterns and the static nature of standard attention, we proposed **DGAFormer**. DGAFormer augments attention with a **Dynamic Graph Attention** operator that injects context-dependent graph updates to capture multi-hop temporal dependencies, and it applies this

operator to both endogenous self-attention and exogenous cross-attention for unified modeling. Moreover, DGAFormer introduces a **Gated Multi-scale Exogenous** Encoder to extract localized exogenous patterns at different temporal resolutions and form informative exogenous representations. Extensive experiments on multiple real-world benchmarks demonstrate that DGAFormer consistently improves forecasting accuracy over strong baselines, with particularly pronounced gains in exogenous-rich settings. Overall, our results highlight the importance of combining dynamic dependency modeling with multi-scale exogenous representation learning for exogenous-aware time series forecasting.

ACKNOWLEDGMENT

This work is supported by Smart Grid-National Science and Technology Major Project 2024ZD0802200.

REFERENCES

[1] A. Menati, F. Doudi, D. Kalathil, and L. Xie, "Powermamba: A deep state space model and comprehensive benchmark for time series prediction in electric power systems," *IEEE Transactions on Power Systems*, 2025.

[2] N. Kim, H. Park, J. Lee, and J. K. Choi, "Short-term electrical load forecasting with multidimensional feature extraction," *IEEE Transactions on Smart Grid*, vol. 13, no. 4, pp. 2999–3013, 2022.

[3] Y. Lv, Y. Duan, W. Kang *et al.*, "Traffic flow prediction with big data: A deep learning approach," *Ieee transactions on intelligent transportation systems*, vol. 16, no. 2, pp. 865–873, 2014.

[4] J. Zhang, J. Tang, J. Jin, and Z. Qu, "Spatio-temporal pre-training enhanced fast pure tansformer network for traffic flow forecasting," in *2023 International Joint Conference on Neural Networks*, 2023, pp. 1–8.

[5] A. Das, W. Kong, A. a. Leach *et al.*, "Long-term forecasting with tide: Time-series dense encoder," *arXiv preprint arXiv:2304.08424*, 2023.

[6] K. G. Olivares, C. Challu, G. Marcjasz, R. Weron, and A. Dubrawski, "Neural basis expansion analysis with exogenous variables: Forecasting electricity prices with nbeatsx," *International Journal of Forecasting*, vol. 39, no. 2, pp. 884–900, 2023.

[7] Y. Wang, H. Wu, J. Dong *et al.*, "Timexer: Empowering transformers for time series forecasting with exogenous variables," *Advances in Neural Information Processing Systems*, vol. 37, pp. 469–498, 2024.

[8] H. Zhou, S. Zhang, J. Peng *et al.*, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *AAAI conference on artificial intelligence*, 2021, pp. 11 106–11 115.

[9] H. Wu, J. Xu, J. Wang, and M. Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *Advances in neural information processing systems*, vol. 34, pp. 22 419–22 430, 2021.

[10] B. M. Williams, "Multivariate vehicular traffic flow prediction: Evaluation of arimax modeling," *Transportation Research Record*, vol. 1776, no. 1, pp. 194–200, 2001.

[11] S. I. Vagropoulos, G. Chouliaras, E. G. Kardakos *et al.*, "Comparison of sarimax, sarima, modified sarima and ann-based models for short-term pv generation forecasting," in *2016 IEEE international energy conference*, 2016, pp. 1–6.

[12] A. Vaswani, N. Shazeer, N. Parmar *et al.*, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[13] R. Li, T. Li, S. Ye *et al.*, "Enhancing generalizability via utilization of unlabeled data for occupancy perception," in *AAAI Conference on Artificial Intelligence*, 2025, pp. 4896–4904.

[14] Z. Song, K. Chen, P. Wang *et al.*, "Unsupervised action segmentation via multi-scale temporal-interaction enhancement," *IEEE Transactions on Circuits and Systems for Video Technology*, 2026.

[15] R. Li, H. Zheng, Z. Yin *et al.*, "Spoformer: Enhancing prior learning for multi-view 3d occupancy perception via semantic-aware attention," in *European Conference on Artificial Intelligence*, 2025.

[16] K. Chen, W. Luo, S. Liu *et al.*, "Powerformer: A section-adaptive transformer for power flow adjustment," in *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2025, pp. 2204–2215.

[17] B. Lim, S. Ö. Arık, N. Loeff, and T. Pfister, "Temporal fusion transformers for interpretable multi-horizon time series forecasting," *International journal of forecasting*, vol. 37, no. 4, pp. 1748–1764, 2021.

[18] Y. Zhang and J. Yan, "Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting," in *International Conference on Learning Representations*, 2023.

[19] A. Yunita, M. I. Pratama, M. Z. Almuzakki *et al.*, "Performance analysis of neural network architectures for time series forecasting: A comparative study of rnn, lstm, gru, and hybrid models," *MethodsX*, vol. 15, p. 103462, 2025.

[20] S. Bai, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[21] T. Zhou, Z. Ma, Q. Wen *et al.*, "Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting," in *International conference on machine learning*. PMLR, 2022, pp. 27 268–27 286.

[22] Z. Dai, Z. Yang, Y. Yang *et al.*, "Transformer-xl: Attentive language models beyond a fixed-length context," in *The 57th annual meeting of the association for computational linguistics*, 2019, pp. 2978–2988.

[23] J. W. Rae, A. Potapenko, S. M. Jayakumar, and T. P. Lillicrap, "Compressive transformers for long-range sequence modelling," *arXiv preprint arXiv:1911.05507*, 2019.

[24] K. Choromanski, V. Likhosherstov, D. Dohan *et al.*, "Rethinking attention with performers," *arXiv preprint arXiv:2009.14794*, 2020.

[25] S. Wang, B. Z. Li, M. Khabsa *et al.*, "Linformer: Self-attention with linear complexity," *arXiv preprint arXiv:2006.04768*, 2020.

[26] K. Chen, J. Song, S. Liu *et al.*, "Distribution knowledge embedding for graph pooling," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 8, pp. 7898–7908, 2023.

[27] K. Chen, S. Liu, T. Zhu *et al.*, "Improving expressivity of gnns with subgraph-specific factor embedded normalization," in *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023, pp. 237–249.

[28] N. Yu, Y. Deng, S. Liu *et al.*, "Disentangled table-graph representation for interpretable transmission line fault location," in *AAAI Conference on Artificial Intelligence*, 2025, pp. 977–985.

[29] L. Han, K. Chen, L. Zhao *et al.*, "Cross-domain animal pose estimation with skeleton anomaly-aware learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 35, no. 9, pp. 9148–9160, 2025.

[30] M. Jin, H. Y. Koh, Q. Wen *et al.*, "A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[31] C. Chen, W. Li, X. Meng *et al.*, "A network latency prediction method based on sequence decomposition and gnn," in *2024 International Joint Conference on Neural Networks*, 2024, pp. 1–7.

[32] Z. Pan, Y. Liang, W. Wang *et al.*, "Urban traffic prediction from spatio-temporal data using deep meta learning," in *ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 1720–1730.

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

[34] C. Song, Y. Lin, S. Guo *et al.*, "Spatial-temporal synchronous graph convolutional networks: A new framework for spatial-temporal network data forecasting," in *AAAI conference on artificial intelligence*, 2020, pp. 914–921.

[35] L. Ø. Bentsen, N. D. Warakagoda *et al.*, "Spatio-temporal wind speed forecasting using graph networks and novel transformer architectures," *Applied Energy*, vol. 333, p. 120565, 2023.

[36] J. Lago, G. Marcjasz, B. De Schutter, and R. Weron, "Forecasting day-ahead electricity prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark," *Applied Energy*, vol. 293, p. 116983, 2021.

[37] M. Liu, A. Zeng, M. Chen *et al.*, "Scinet: Time series modeling and forecasting with sample convolution and interaction," *Advances in Neural Information Processing Systems*, vol. 35, pp. 5816–5828, 2022.