STENet: A Soft Translation-Equivariant Network for Time Series Forecasting

1st Dongwei Liu
South China Normal University
Guangzhou, China
liudongwei@m.scnu.edu.cn

2nd Weiping Zheng*
South China Normal University
Guangzhou, China
zhengweiping@scnu.edu.cn

3rd Huande Liu
South China Normal University
Guangzhou, China
lhd0423@126.com

***Abstract*—Time series forecasting plays a critical role in a wide range of real-world applications. However, translation equivariance remains an important yet relatively underexplored property in this field. Existing forecasting models typically neglect this property and are therefore forced to implicitly learn equivariant behavior from data, resulting in reduced model capacity. In this paper, we propose STENet, a novel forecasting architecture that explicitly encodes translation equivariance via a novel point-wise accumulation prediction paradigm, while adaptively relaxing the equivariance constraint when it is violated. Extensive experiments on 11 real-world multivariate time series datasets demonstrate that STENet achieves competitive performance with state-of-the-art methods. Moreover, quantitative analysis using an empirical equivariance metric provides empirical evidence for the effectiveness and usefulness of incorporating adaptive translation equivariance in time series forecasting models.**

***Index Terms*—Time series forecasting, Translation equivariance, Inductive bias.**

I. Introduction

Time series forecasting, as an important task in time series analysis, aims to construct predictive models from historical observations to infer future temporal variations. It plays a critical role in a wide range of real-world applications, including energy scheduling [1] and investment strategy [2]. In recent years, numerous approaches based on deep learning have been proposed for time series forecasting, most of which focus on modeling intrinsic characteristics such as non-stationarity [3], [4], sequential dependency [5], [6], inter-variable correlation [7]–[9], and seasonality [10], [11].

However, a largely overlooked yet essential property of time series forecasting is temporal translation equivariance. As Fig. 1 shows, in the context of time series forecasting, translation equivariance, which is commonly referred to as time invariance [12], is the property that when a historical sequence is shifted along the time axis within the look-back window, the corresponding predicted future sequence should undergo an identical temporal shift. Moreover, the underlying mapping function f that relates the past observations to future values remains invariant under such temporal translation. Most existing time series forecasting models do not explicitly encode this property as an inductive bias. When the input historical sequence is shifted, these models do not inherently produce predictions that are shifted in a synchronized manner,

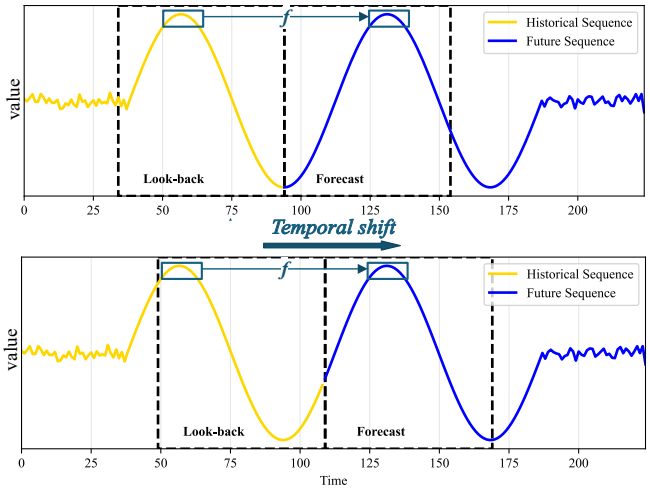

Fig. 1. Illustration of Temporal Translation Equivariance.

forcing the model to implicitly learn this property from data, often at the cost of increased optimization burden. As a consequence, a substantial portion of model capacity is expended on approximating equivariant behavior rather than capturing intrinsic temporal patterns.

On the other hand, while translation-equivariant inductive biases can improve parameter efficiency, enforcing them as hard constraints may be overly restrictive in practice [13], [14]. Limited look-back windows, concept drift, and non-stationarity can cause the optimal mapping to deviate from strict equivariance, leading a strictly equivariant model to misalign predictions and waste capacity. To address this, we introduce a soft equivariance mechanism via a gating module that adaptively modulates historical inputs, enabling the model to exploit translation-equivariant structure where applicable while remaining flexible to deviations.

In this paper, we propose **S**oft **T**ranslation-**E**quivariant **Net**work (**STENet**), a time series forecasting model that includes a module with built-in translation-equivariant inductive bias. The module adopts a novel point-wise accumulation prediction paradigm, where shared parameters are employed to estimate the contribution of each historical data point to its future sequence. The final forecast is then obtained

*Corresponding author.

via a sliding extraction operation over the predicted future sequences. To further relax the equivariant inductive bias, we introduce a gating module that outputs a weight vector, which is used to adaptively modulate historical sequences and accommodate information variation during temporal shift. The main contributions of this work are summarized as follows:

1) We introduce a translation-equivariant forecasting module based on a point-wise accumulation prediction mechanism, which explicitly encodes translation equivariance and enhances model capacity.
2) We develop a Transformer-based gating module that softens the equivariant inductive bias, enabling adaptive modeling of partially translation-equivariant temporal patterns.
3) Extensive experiments on 11 real-world datasets demonstrate that our approach achieves competitive performance compared to state-of-the-art time series forecasting models. In addition, we used a quantitative metric to measure the degree of translation equivariance between models and datasets, providing empirical evidence of the effectiveness of our architectural design.

The code of our model is available at https://github.com/dongweiLiu/STENet/.

II. Related work

The incorporation of equivariance into deep learning architectures has been extensively investigated across a wide range of tasks [15]–[19]. The parameter-sharing principle remains a central and widely adopted mechanism for constructing translation-equivariant neural networks. In time series forecasting, translation equivariance is commonly embedded through autoregressive prediction paradigms in which parameters are shared across time steps, as exemplified by Long Short-Term Memory (LSTM) [20]. However, recursive forecasting strategies inherently suffer from error accumulation [21]. Consequently, such approaches often demonstrate inferior performance in long-term forecasting compared with the direct multi-step prediction mechanism [22].

Despite their empirical effectiveness, many recent multi-step forecasting models largely neglect the translation-equivariant structure intrinsic to temporal prediction. Although translation-equivariant modules such as RNNs or CNNs may be employed within these architectures, they are typically treated merely as sequence-level feature encoders and coupled with non-translation-equivariant prediction schemes [23], [24]. As a result, the model largely loses translation equivariance, having to allocate additional capacity to learn translation-equivariant patterns from data.

III. Preliminary

A. Problem Statement

In this work, we focus on the problem of multivariate time series forecasting. Given a historical sequence $\mathbf{X}_{1:L} \in \mathbb{R}^{M\times L}$ within a look-back window of length L, where M denotes the number of variables, the goal is to predict the subsequent sequence of the next T time steps $\mathbf{X}_{L+1:L+T} \in \mathbb{R}^{M\times T}$. Following the channel-independent setting [5], the multivariate time series is decomposed into M univariate sequences, each of which is processed by a shared model backbone. For simplicity, we describe the following modules using a single univariate sequence $\mathbf{x} \in \mathbb{R}^{L}$.

B. Translation Equivariance

Let $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_L) \in X$ be a univariate time series of length L. A temporal shift by τ steps is defined as

$$T_\tau(\mathbf{x}) = (\mathbf{x}_{1+\tau}, \ldots, \mathbf{x}_{L+\tau}), \tag{1}$$

with appropriate boundary conditions (e.g., zero-padding or circular padding). The set of all shifts forms a group under composition,

$$G = \{T_\tau \mid \tau \in \mathbb{Z}\}, \tag{2}$$

with identity T_0 and inverse $T_{-\tau}$. A forecasting model $f : X \to Y$ is **translation-equivariant** if, for all $\mathbf{x} \in X$ and $T_\tau \in G$,

$$f(T_\tau \mathbf{x}) = T_\tau f(\mathbf{x}), \tag{3}$$

i.e., shifting the input results in an identical shift of the output, formally capturing invariance under temporal translations.

IV. Method

A. Structure Overview

To inherently embed the soft translation equivariance inductive bias for time series forecasting, we design a novel model termed STENet, as illustrated in Fig. 2.

B. Translation-equivariant Prediction Module

Because the contribution of individual time steps is implicitly entangled within the global representation, the conventional direct multi-step prediction mechanism does not explicitly encode translation equivariance. To overcome this limitation, we propose a Translation-Equivariant Prediction Module (TEPM). The key innovation lies in a point-wise accumulation prediction mechanism. Rather than predicting from a single global latent representation, the module leverages parameter-shared operators across time steps to forecast future sequences for every valid data point within the look-back window.

Specifically, a DCCNN (Dilated Causal Convolutional Neural Network), which is composed solely of stacked 1D dilated causal convolutional layers [25], [26], processes the input sequence $\tilde{\mathbf{x}} \in \mathbb{R}^{L}$ to extract hierarchical temporal features of each data point:

$$\mathbf{H} = \mathrm{DCCNN}(\tilde{\mathbf{x}}), \tag{4}$$

where $\mathbf{H} \in \mathbb{R}^{K\times D}$ is the resulting feature representation, D denotes the number of channels in the final convolutional layer, and K is the output length. All convolutional layers adopt valid convolution and a stride of 1. Valid convolution mitigates artificial boundary effects and reduces computational overhead, while a unit stride preserves the translation-equivariant inductive bias.

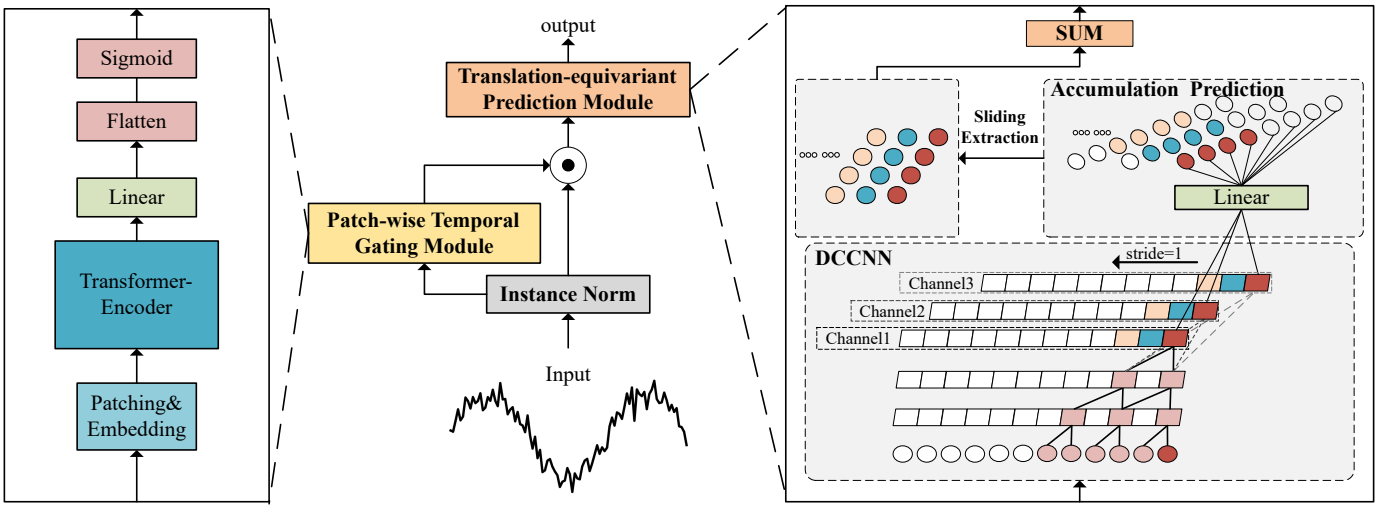

Fig. 2. Overall architecture of STENet. The process begins by applying Instance Normalization to the input time series $\mathbf{x}$ to stabilize the data distribution. Then the Translation-equivariant Prediction Module (TEPM) incorporates a solid inductive bias of translation equivariance through a point-wise accumulation prediction paradigm. Complementarily, the Patch-wise Temporal Gating Module (PTGM) introduces a gating strategy that adaptively relaxes the model's overall translation equivariance.

Next, a linear projection layer maps each feature vector to a corresponding relative future sequence of length $S = T + (K-1)$:

$$\mathbf{R} = \text{Linear}(\mathbf{H}), \tag{5}$$

where $\mathbf{R} \in \mathbb{R}^{K \times S}$.

To construct the final forecasting result, a sliding window extraction operation is applied. Specifically, the final prediction $\hat{\mathbf{y}} \in \mathbb{R}^T$ is obtained by aggregating contributions from corresponding data points. Formally,

$$\hat{\mathbf{y}} = \sum_{k=0}^{K-1} \mathbf{R}_{K-1-k}[k : k+T-1]. \tag{6}$$

When the data points in the input sequence $\mathbf{x}$ undergo a temporal shift, the shared convolutional kernels and linear projection layers ensure that the corresponding components in $\mathbf{H}$ and $\mathbf{R}$ remain invariant. The subsequent extraction and summation operations guarantee that the contribution of each data point to the final forecast shifts synchronously with the input.

C. Patch-wise Temporal Gating Module

Although the TEPM inherently embeds a translation-equivariant inductive bias, it exhibits two key limitations. In practice, translation equivariance holds only approximately and within limited regimes, rather than as a universal property. Moreover, the effective receptive field of the DCCNN is finite, which constrains its ability to capture long-range dependencies across distant time steps.

To address these limitations, we propose a Patch-wise Temporal Gating Module (PTGM), inspired by the LSTM gating mechanism [20]. The PTGM generates a learnable point-wise gating vector $\mathbf{w} \in \mathbb{R}^L$ to adaptively modulate the input sequence, selectively relaxing rigid equivariance constraints when necessary.

Given the normalized input $\mathbf{x}_{\text{norm}} \in \mathbb{R}^L$, we follow the prior work [5] and partition it into $N = \lfloor L/P \rfloor$ non-overlapping patches of length P. Each patch is then linearly projected into a d-dimensional embedding, yielding $\mathbf{E} \in \mathbb{R}^{N \times d}$. A standard Transformer encoder is subsequently applied to produce contextualized representations, which are projected back to patch-wise weight vectors via a linear layer with dropout:

$$\mathbf{G}_p = \text{Dropout}(\text{Linear}(\mathbf{Z})) \in \mathbb{R}^{N \times P}. \tag{7}$$

Flattening $\mathbf{G}_p$ and applying a sigmoid yields the gating vector:

$$\mathbf{w} = \sigma(\text{Flatten}(\mathbf{G}p)), \tag{8}$$

which modulates the input sequence element-wise:

$$\tilde{\mathbf{x}} = \mathbf{x}_{\text{norm}} \odot \mathbf{w}. \tag{9}$$

The weighted sequence $\tilde{\mathbf{x}}$ is then fed into the TEPM to generate the final forecast $\hat{\mathbf{y}}$.

V. Experiments

A. Experimental Settings

a) Dataset: We evaluated STENet on 11 popular time series datasets from various domains. The detailed characteristics of the datasets are summarized in Table I. In particular, the value of the Transition [27] metric is obtained from TFB [28]. This metric is used to quantify the extent of fixed and recognizable structural patterns in a dataset, such as trends, periodicity, and seasonality. The smaller the Transition value, the stronger this phenomenon.

TABLE I
STATISTICS OF MULTIVARIATE DATASETS.

Dataset	Domain	Lengths	Dim	Split	Transition
ETTh1	Electricity	14,400	7	6:2:2	0.020
ETTm1	Electricity	57,600	7	6:2:2	0.027
ETTh2	Electricity	14,400	7	6:2:2	0.042
ETTm2	Electricity	57,600	7	6:2:2	0.038
Weather	Environment	52,696	21	7:1:2	0.037
Electricity	Electricity	26,304	321	7:1:2	0.011
Traffic	Traffic	17,856	862	7:1:2	0.011
ILI	Health	966	1	7:1:2	0.038
NN5	Banking	791	11	7:1:2	0.007
Wike2000	Web	1,792	2,000	7:1:2	0.037
Covid-19	Health	1,392	948	7:1:2	0.126

b) Baseline: We compare our approach with state-of-the-art models, including MLP-based models (DTAF [3], TimeMixer [29], and DLinear [22]), CNN-based models (xPatch [11], MICN [24], and TimesNet [23]), Transformer-based models (iTransformer [8], and PatchTST [5]).

c) Implementation Details: All baselines are implemented and evaluated under the unified TFB [28] benchmark to guarantee fair and consistent comparisons. We select Mean Absolute Error (MAE) and Mean Squared Error (MSE) as evaluation metrics. The datasets ILI, Covid-19, NN5, and Wike2000 are evaluated with horizons {24, 36, 48, 60}, while the remaining datasets use {96, 192, 336, 720}. The look-back window sizes are set to {36, 104} for ILI, Covid-19, NN5, and Wike2000, {96, 336, 512} for other datasets, and each baseline's best result under these settings is reported.

B. Main Results

Table II reports the averaged performance of our model. STENet achieves the best MSE on 8 out of 11 datasets and the best MAE on 9 out of 11 datasets. Compared with the recent state-of-the-art method DTAF, STENet consistently attains comparable or superior results across most datasets. Furthermore, it substantially outperforms earlier models with different architectures; for instance, on the Covid-19 dataset, STENet improves MAE by 16.6% and MSE by 28.5% over PatchTST. Overall, these results demonstrate the effectiveness of the proposed design and establish STENet as a highly competitive model for multivariate time series forecasting.

C. Ablation Study

To evaluate the effectiveness of our key module designs, we conduct the following three sets of ablation experiments by removing specific modules or properties:

*a) **w/o TE**:* To verify the effectiveness of the embedded translation equivariance inductive bias, we replace the point-wise accumulation prediction mechanism in TEPM by projecting the input sequence's feature representation directly onto the prediction outputs via a fully connected layer.

*b) **w/o PTGM**:* We remove the PTGM and omit the modulation on the historical sequences, which leads to a more constrained equivariant design.

*c) **Simple Weighting**:* We replace the PTGM-generated weight vector with a learnable weight that, once trained, remains fixed and does not adapt to variations in the historical time series.

The ablation results are summarized in Table III. We observe that removing the translation equivariance inductive bias leads to a consistent performance degradation across all datasets. This indicates the importance of building in translation equivariance as an inductive bias for the forecasting task.

In addition, the removal of the PTGM and the simple weighting also result in a universal performance degradation across all benchmarks. This indicates that our design of soft translation equivariance is superior to enforcing a strict equivariance constraint.

D. Translation-Equivariance Evaluation

To quantitatively assess the translation equivariance of various time series forecasting models, we adopt the G-Empirical Equivariance Deviation (G-EED) metric [30]. This metric provides a rigorous, direct measure of the extent to which a model deviates from perfect G-equivariance on a given dataset $\mathcal{D}$. It allows for a comparison of how different forecasting models capture or fail to capture the translation equivariance.

Let $G = \{T_\tau \mid \tau \in \{1, 2, \ldots, 32\}\}$ be a finite subset of the integer translation group, where each T_τ denotes a temporal shift by τ steps. For a forecasting model $\mathcal{F} : \mathbb{R}^L \to \mathbb{R}^T$, the G-EED for the model $\mathcal{F}$ is defined as

$$\mathcal{E}(\mathcal{F}, G) = \frac{1}{|\mathcal{D}||G|} \sum_{\mathbf{x} \in \mathcal{D}} \sum_{T_\tau \in G} m\Big(\mathcal{F}(T_\tau \cdot \mathbf{x}),\ T_\tau \cdot \mathcal{F}(\mathbf{x})\Big), \quad (10)$$

where m is the distance function. A model that is perfectly translation-equivariant would yield $\mathcal{E}(\mathcal{F}, G) = 0$, whereas larger values indicate a greater deviation from equivariance.

For evaluation, we employ the standard Euclidean distance as the metric. The test set $\mathcal{D}$ uses a look-back window of length 512 and a prediction horizon of 336. Both $\mathcal{F}(T_\tau \cdot \mathbf{x})$ and $T_\tau \cdot \mathcal{F}(\mathbf{x})$ are Z-score normalized to prevent the distance measure from being biased by differences in data scale.

a) Comparison with different baselines: As shown in Fig. 3, across all datasets, the various baselines exhibit high G-EED values when untrained, which are close to the expected Euclidean distance between two normalized one-dimensional random vectors of length 336. This indicates that these models essentially lack translation equivariance at this stage. After training, the G-EED values of these baselines decrease significantly, suggesting that they have learned translation equivariance from the data during training. In contrast, our model demonstrates notably lower G-EED values than other baselines even before training, showing that it inherently incorporates a certain level of translation equivariance as an inductive bias. Although Fig. 3 shows only a subset due to space limitations, similar trends are generally observed across the remaining models and datasets, with minor variations.

b) Correlation between Transition and Translation Equivariance: As illustrated in the scatter plot in Fig. 4, the

TABLE II

MULTIVARIATE TIME SERIES FORECASTING RESULTS. RESULTS ARE AVERAGED FROM ALL FORECASTING HORIZONS. THE BEST RESULTS ARE HIGHLIGHTED IN **BOLD** AND THE SECOND-BEST RESULTS ARE UNDERLINED.

Models	STENet (Ours)		DTAF (2026)		xPatch (2025)		TimeMixer (2024)		iTransformer (2024)		PatchTST (2023)		DLinear (2023)		MICN (2023)		TimesNet (2023)	
Metrics	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	**0.399**	**0.412**	**0.399**	0.418	0.415	0.425	0.427	0.441	0.439	0.448	0.419	0.436	0.425	0.439	0.420	0.447	0.468	0.459
ETTh2	**0.338**	**0.379**	0.339	**0.379**	0.343	0.384	0.349	0.397	0.370	0.403	0.351	0.395	0.470	0.468	0.482	0.472	0.390	0.417
ETTm1	**0.340**	**0.364**	0.343	0.368	0.354	0.375	0.356	0.380	0.361	0.390	0.349	0.381	0.356	0.378	0.355	0.383	0.408	0.415
ETTm2	**0.252**	**0.306**	0.253	0.308	**0.252**	0.309	0.257	0.318	0.269	0.327	0.256	0.314	0.259	0.324	0.294	0.357	0.292	0.331
Weather	**0.222**	**0.250**	**0.222**	0.251	0.225	0.252	0.226	0.264	0.232	0.270	0.224	0.262	0.242	0.293	0.239	0.289	0.255	0.282
Electricity	0.160	**0.247**	0.160	0.248	**0.158**	0.248	0.185	0.284	0.163	0.258	0.171	0.270	0.167	0.264	0.179	0.290	0.190	0.290
ILI	1.817	0.850	**1.689**	**0.801**	2.017	0.921	1.820	0.886	1.857	0.892	1.902	0.879	2.185	1.040	2.368	1.049	2.164	0.934
Traffic	0.406	**0.246**	0.403	0.249	0.404	0.261	0.409	0.279	**0.397**	0.281	**0.397**	0.275	0.418	0.287	0.539	0.313	0.617	0.327
NN5	**0.639**	**0.529**	0.644	0.538	0.766	0.611	0.656	0.556	0.660	0.550	0.689	0.590	0.692	0.580	0.669	0.569	0.684	0.567
Wike2000	**486.038**	1.127	491.354	**1.124**	511.319	1.177	518.679	1.251	545.647	1.190	513.964	1.140	630.769	1.394	587.885	1.613	527.790	1.249
Covid-19	**1.339**	**0.040**	1.352	0.041	1.705	0.061	12.941	0.133	1.488	0.050	1.607	0.056	8.074	0.239	52.116	0.535	2.609	0.077
1st Count	**8**	**9**	3	3	2	0	0	0	1	0	1	0	0	0	0	0	0	0

TABLE III

ABLATION STUDY. RESULTS ARE AVERAGED OVER ALL FORECASTING HORIZONS. BEST RESULTS ARE SHOWN IN **BOLD**.

Models	STENet		w/o TE		w/o PTGM		Simple Weighting	
Metrics	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	**0.399**	**0.412**	0.444	0.447	0.406	0.419	0.407	0.419
ETTh2	**0.338**	**0.379**	0.381	0.410	0.346	0.385	0.351	0.388
ETTm1	**0.340**	**0.364**	0.367	0.383	0.351	0.369	0.353	0.370
ETTm2	**0.252**	**0.306**	0.259	0.313	0.256	0.310	0.245	0.309
Weather	**0.222**	**0.250**	0.230	0.260	0.246	0.274	0.255	0.272
Electricity	**0.160**	**0.247**	0.176	0.262	0.168	0.258	0.167	0.257
NN5	**0.639**	**0.529**	0.707	0.584	0.685	0.563	0.703	0.575

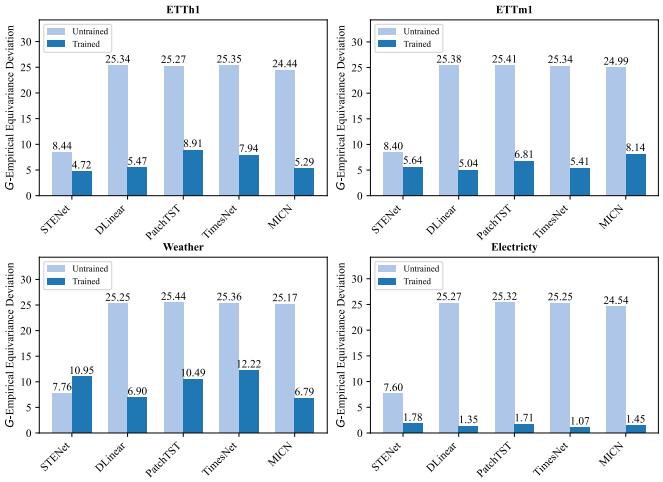

Fig. 3. Comparison of G-EED metrics across models and datasets. "Untrained" refers to randomly initialized parameters, while "Trained" denotes the model after training on the corresponding training set.

results reveal a strong correlation between the dataset's Transition and our model's G-EED (after training). When the model is able to accurately capture the mapping between inputs and outputs, it tends to preserve translation equivariance. Conversely, for datasets with weak regularity or low predictability, the model exhibits a tendency to break translation equivariance. This suggests that translation equivariance emerges as a data-dependent property and highlights the necessity of the proposed PTGM, which dynamically adjusts the strength of translation equivariance.

E. Model Efficiency

Table IV reports the parameter counts of four time-series forecasting models on the ETTh1 dataset across prediction horizons of {96, 192, 336, 720}. The proposed model consistently achieves the smallest parameter size under all horizons, demonstrating its parameter efficiency.

TABLE IV

PARAMETER COUNTS (MILLIONS) OF DIFFERENT MODELS ON ETTH1.

Models	**STENet**	**DTAF**	**xPatch**	**TimesNet**	**TimeMixer**
96	**0.048**	0.421	0.469	0.605	1.649
192	**0.053**	0.800	1.043	0.614	1.741
336	**0.060**	1.404	2.129	0.628	1.102
720	**0.178**	2.972	5.327	0.666	2.250

VI. CONCLUSIONS

In this work, we investigated translation equivariance as a fundamental yet often overlooked inductive bias in time

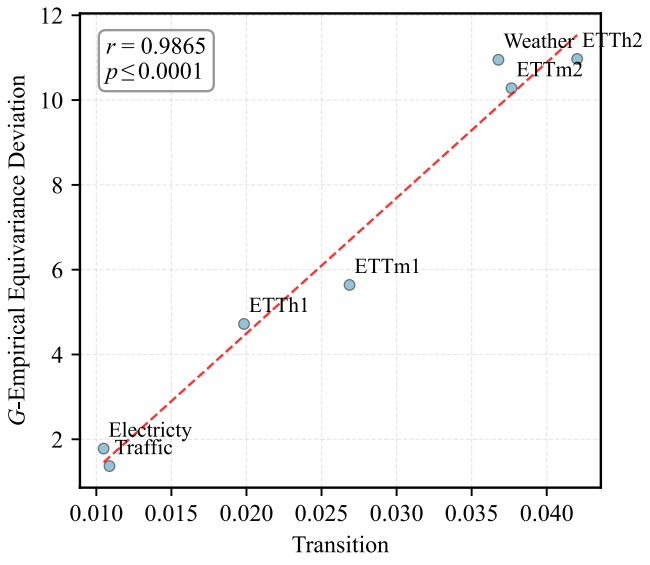

Fig. 4. Relationship between the dataset's Transition and the G-EED of our model trained on the corresponding dataset (dashed red line: fitted linear regression; r: Pearson correlation coefficient; p: associated p-value).

series forecasting. To explicitly incorporate this property, we proposed a forecasting framework, STENet. It integrates a Translation-equivariant Prediction Module (TEPM) that inherently enforces translation equivariance by adopting a novel point-wise accumulation prediction mechanism. Furthermore, we introduce a Patch-wise Temporal Gating Module (PTGM), which adaptively relaxes this constraint. It enables the model to better accommodate complex time series characteristics in reality. Based on these, STENet achieves competitive forecasting performance across diverse datasets. In addition, we introduce an empirical quantitative metric to assess the degree of translation equivariance exhibited by different models, which provides direct evidence supporting the effectiveness of the proposed architectural design.

REFERENCES

[1] H. Xu, F. Hu, X. Liang, G. Zhao, and M. Abugunmi, "A framework for electricity load forecasting based on attention mechanism time series depthwise separable convolutional neural network," *Energy*, vol. 299, p. 131258, 2024.

[2] Z. Zhu, H. Chen, Q. Qu, and V. Chung, "Fincast: A foundation model for financial time-series forecasting," in *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, 2025, pp. 4539–4549.

[3] J. Lu, P. Chen, C. Guo, Y. Shu, M. Wang, and B. Yang, "Towards non-stationary time series forecasting with temporal stabilization and frequency differencing," in *AAAI*, 2026.

[4] T. Dai, B. Wu, P. Liu, N. Li, X. Yuerong, S.-T. Xia, and Z. Zhu, "Ddn: dual-domain dynamic normalization for non-stationary time series forecasting," in *NeurIPS*, 2024.

[5] Y. Nie, N. H. Nguyen, P. Sinthong, and J. Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," in *ICLR*, 2023.

[6] T. Dai, B. Wu, P. Liu, N. Li, J. Bao, Y. Jiang, and S.-T. Xia, "Periodicity decoupling framework for long-term series forecasting," in *ICLR*, 2024.

[7] S. Lin, H. Chen, H. Wu, C. Qiu, and W. Lin, "Temporal query network for efficient multivariate time series forecasting," in *International Conference on Machine Learning*, 2025.

[8] Y. Liu, T. Hu, H. Zhang, H. Wu, S. Wang, L. Ma, and M. Long, "itransformer: Inverted transformers are effective for time series forecasting," in *ICLR*, 2023.

[9] X. Qiu, X. Wu, Y. Lin, C. Guo, J. Hu, and B. Yang, "Duet: Dual clustering enhanced multivariate time series forecasting," in *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1*, 2025, pp. 1185–1196.

[10] S. Huang, Z. Zhao, C. Li, and L. BAI, "Timekan: Kan-based frequency decomposition learning architecture for long-term time series forecasting," in *ICLR*, 2025.

[11] A. Stitsyuk and J. Choi, "xpatch: Dual-stream time series forecasting with exponential seasonal-trend decomposition," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 19, 2025, pp. 20 601–20 609.

[12] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, 3rd ed., ser. Pearson Education Signal Processing Series. Pearson, 2009.

[13] J. J. Dabrowski, Y. Zhang, and A. Rahman, "Forecastnet: A time-variant deep feed-forward neural network architecture for multi-step-ahead time-series forecasting," in *International conference on neural information processing*. Springer, 2020, pp. 579–591.

[14] R. Wang, R. Walters, and R. Yu, "Data augmentation vs. equivariant networks: A theory of generalization on dynamics forecasting," *arXiv preprint arXiv:2206.09450*, 2022.

[15] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities." *Proceedings of the national academy of sciences*, vol. 79, no. 8, pp. 2554–2558, 1982.

[16] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 2002.

[17] J. Hu, Z. Yao, L. Jin, H. He, and Y. Lu, "Enhancing image restoration transformer via adaptive translation equivariance," *arXiv preprint arXiv:2506.18520*, 2025.

[18] J. Gordon, W. P. Bruinsma, A. Y. Foong, J. Requeima, Y. Dubois, and R. E. Turner, "Convolutional conditional neural processes," in *ICLR*, 2020.

[19] I. Bello, B. Zoph, A. Vaswani, J. Shlens, and Q. V. Le, "Attention augmented convolutional networks," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 3286–3295.

[20] A. Graves, "Long short-term memory," *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012.

[21] B. Lim and S. Zohren, "Time-series forecasting with deep learning: a survey," *Philosophical transactions of the royal society a: mathematical, physical and engineering sciences*, vol. 379, no. 2194, 2021.

[22] A. Zeng, M. Chen, L. Zhang, and Q. Xu, "Are transformers effective for time series forecasting?" in *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, no. 9, 2023, pp. 11 121–11 128.

[23] H. Wu, T. Hu, Y. Liu, H. Zhou, J. Wang, and M. Long, "Timesnet: Temporal 2d-variation modeling for general time series analysis," in *ICLR*, 2023.

[24] H. Wang, J. Peng, F. Huang, J. Wang, J. Chen, and Y. Xiao, "Micn: Multi-scale local and global context modeling for long-term series forecasting," in *ICLR*, 2023.

[25] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, K. Kavukcuoglu *et al.*, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

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

[27] C. H. Lubba, S. S. Sethi, P. Knaute, S. R. Schultz, B. D. Fulcher, and N. S. Jones, "catch22: Canonical time-series characteristics: Selected through highly comparative time-series analysis," *Data mining and knowledge discovery*, vol. 33, no. 6, pp. 1821–1852, 2019.

[28] X. Qiu, J. Hu, L. Zhou, X. Wu, J. Du, B. Zhang, C. Guo, A. Zhou, C. S. Jensen, Z. Sheng *et al.*, "Tfb: Towards comprehensive and fair benchmarking of time series forecasting methods," *Proceedings of the VLDB Endowment*, vol. 17, no. 9, pp. 2363–2377, 2024.

[29] S. Wang, H. Wu, X. Shi, T. Hu, H. Luo, L. Ma, J. Y. Zhang, and J. Zhou, "Timemixer: Decomposable multiscale mixing for time series forecasting," in *ICLR*, 2024.

[30] H. Kvinge, T. H. Emerson, G. Jorgenson, S. Vasquez, T. Doster, and J. D. Lew, "In what ways are deep neural networks invariant and how should we measure this?" in *NeurIPS*, 2022, pp. 32 816–32 829.