

Forecasting with Guidance: Representation-Level Supervision for Time Series Forecasting

Jiacheng Wang

Xijing University

2408540402040@stu.xijing.edu.cn

Liang Fan

Loughborough University

L.Fan@lboro.ac.uk

Baihua Li

Loughborough University

B.Li@lboro.ac.uk

Luyan Zhang*

Northeastern University

zhang.luya@northeastern.edu

Abstract—Nowadays, time series forecasting is predominantly approached through the end-to-end training of deep learning architectures using error-based objectives. While this is effective at minimizing average loss, it encourages the encoder to discard informative yet extreme patterns. This results in smooth predictions and temporal representations that poorly capture salient dynamics. To address this issue, we propose ReGuider, a plug-in method that can be seamlessly integrated into any forecasting architecture. ReGuider leverages pretrained time series foundation models as semantic teachers. During training, the input sequence is processed together by the target forecasting model and the pretrained model. Rather than using the pretrained model’s outputs directly, we extract its intermediate embeddings, which are rich in temporal and semantic information, and align them with the target model’s encoder embeddings through representation-level supervision. This alignment process enables the encoder to learn more expressive temporal representations, thereby improving the accuracy of downstream forecasting. Extensive experimentation across diverse datasets and architectures demonstrates that our ReGuider consistently improves forecasting performance, confirming its effectiveness and versatility.

Index Terms—Time Series forecasting, Foundation Model, Representation Learning

I. INTRODUCTION

Time series forecasting (TSF) is central to many real-world applications, including finance [1], healthcare [2], and climate science [3]. The recent success of deep learning has brought substantial advances to the field, with architectures such as graph networks [4], [5], Linear-based models [6]–[8], and transformers [9], [10] demonstrating strong predictive capabilities. By automatically extracting complex temporal dependencies, deep learning models have surpassed classical statistical approaches and become the predominant choice for modern forecasting tasks. However, achieving accurate and robust predictions across diverse domains remains a fundamental challenge.

Most deep learning approaches [11]–[16] to time series forecasting rely solely on error-based objectives such as mean squared error (MSE) and mean absolute error (MAE). While these objectives optimize predictive accuracy directly, they provide the encoder with limited guidance on how to capture rich temporal dependencies. Consequently, models often reduce errors by averaging predictions, which can result in the neglect of outlier events and the formation of overly smoothed representations [17]. This issue is particularly evident in the

learned embeddings, which fail to encode sufficient temporal semantics. Such latent representations are often “semantically impoverished” as they capture the trend but lose the underlying generative dynamics of the system.

We argue that the key to improving forecasting performance lies not in designing increasingly complex architectures, but in incorporating external semantic supervision. Moreover, time series foundation models [18]–[20], trained on large-scale and diverse data, learn temporal representations that more faithfully capture fine-grained and semantically rich patterns. To address the limitations of error-only supervision, we propose explicitly guiding the encoder using representations extracted from time-series foundation models, enabling it to learn more meaningful temporal abstractions. The core idea is to enrich the embeddings of forecasting methods through external semantic supervision, thereby enhancing their representational capacity without increasing architectural complexity.

Technically, we propose ReGuider, a representation-level supervision plug-in designed to enhance time series forecasting. The central concept involves leveraging pretrained time series foundation models as semantic teachers. During training, both the target forecasting model and the pretrained model process the same input sequence. Instead of using the pretrained model’s prediction head, we extract its intermediate embeddings, which encode rich temporal dependencies and semantic structures. These embeddings are then aligned with the encoder representations of the target model, thereby encouraging the encoder to learn more expressive and temporally coherent representations. ReGuider is model-agnostic, enabling it to be seamlessly integrated into a wide range of TSF methods without altering their original structure.

In summary, this work makes the following contributions:

- We identify the limitation of error-only supervision in deep learning-based forecasting and propose to enhance temporal embeddings through external semantic guidance.
- We develop ReGuider, a plug-in method that aligns encoder representations with pretrained time series foundation models, enriching the temporal semantics of learned embeddings.
- We conduct extensive experiments across diverse datasets and architectures, demonstrating that ReGuider consistently improves forecasting accuracy and generalizes effectively to different backbone models.

*Corresponding to Luyan Zhang.

II. BACKGROUND AND RELATED WORK

A. Deep Learning in Time Series Forecasting

Deep learning [21], [22] has become the dominant paradigm in time series forecasting, with RNNs, CNNs, GNNs, and Transformers all demonstrating strong empirical results [12], [23]–[26]. Recently, the community has started training large scale time series foundation models using hierarchical transformers or masked autoencoders that have been pre-trained on millions of sequences [18]–[20], [27]. The immense capacity and extensive pre-training of these models enable them to capture universal temporal dynamics, ranging from short-term seasonality to long-term trends. Current practice involves either freezing or lightly fine-tuning these models for direct prediction or few-shot adaptation. However, we exploit their internal representations as repositories of high-quality temporal knowledge to enhance any downstream forecaster.

B. Representation Learning

Representation learning is essential for enabling models to capture informative and transferable features. Across domains, it has been utilized to impose inductive biases that extend beyond simple task losses. For instance, in diffusion models, it is applied to make noise patterns more structured and controllable, improving generation quality and stability [28]. In vision and language, aligning latent representations with pretrained models has proven effective in enriching feature spaces and boosting downstream performance [29].

Within time series forecasting, traditionally, TSF models used supervised encoders to extract features for point-wise prediction [6], [12], [30]–[32]. However, these are prone to “representation collapse” when driven solely by MSE, filtering out critical regime shifts to minimize average loss. While self-supervised learning (SSL) and contrastive paradigms attempt to mitigate this, they often rely on heuristic augmentations and remain limited by the scale of individual datasets.

The emergence of Time Series Foundation Models (TSFMs) has redefined this landscape. Pretrained on billions of sequences, TSFMs [18], [20], [33]–[35] develop a “universal temporal vocabulary” that captures nuanced seasonality and structural dependencies. ReGuider bridges the gap between these high-capacity models and efficient task-specific predictors by using TSFM embeddings as a “semantic gold standard” for alignment. This representation-level supervision can highlight long-term seasonality, abrupt regime shifts, and inter-variable relations that are often overlooked by error-driven objectives, thereby leading to more accurate and robust predictions.

III. METHOD

A. Problem Statement

The goal of TSF is to predict a future sequence $Y \in \mathbb{R}^{C \times T}$ with horizon T from a past sequence $X \in \mathbb{R}^{C \times L}$ of length L , where C denotes the number of variables.

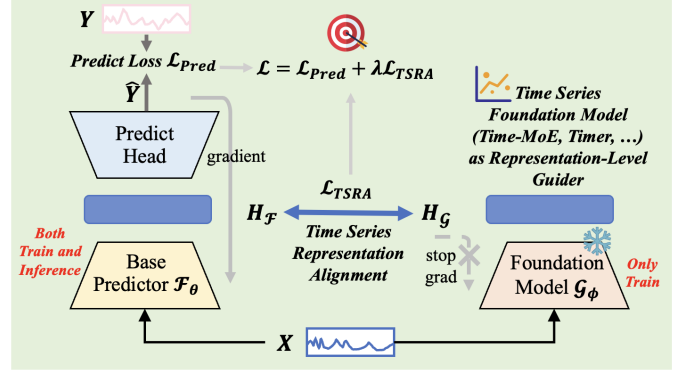


Fig. 1. Overall architecture of ReGuider, which consists of the base predictor $\mathcal{F}_\theta(\cdot)$ and the foundation model $\mathcal{G}_\phi(\cdot)$, serving as a representation guide.

B. Architecture of ReGuider

As shown in Fig. 1, ReGuider is designed to improve time series forecasting by enriching encoder representations through supervision with pre-trained foundation models. Specifically, given an input sequence $X \in \mathbb{R}^{C \times L}$, it is processed through two parallel pathways: (1) the base predictor $\mathcal{F}_\theta(\cdot)$, representing the forecasting model to be trained, and (2) the foundation model $\mathcal{G}_\phi(\cdot)$, serving as a representation guide. The base predictor $\mathcal{F}_\theta(\cdot)$ encodes X into a latent representation H_f before passing it to the prediction head to generate an estimate of the target variable, denoted by Y . For the guider, rather than using the final prediction output of $\mathcal{G}_\phi(\cdot)$, we extract its intermediate embedding H_g from the encoder. This captures the rich temporal patterns and high-level semantics learned during large-scale pretraining. We then introduce a representation supervision objective to minimise the distance between H_f and H_g . This guides the encoder of the base predictor to incorporate the temporal dependencies and semantic structures present in the pretrained $\mathcal{G}_\phi(\cdot)$.

This supervision is seamlessly integrated into the training process alongside the standard forecasting loss, enabling the model to learn to minimize predictive error and produce embeddings that align with stronger temporal representation space simultaneously. This framework is also model-agnostic. ReGuider does not alter the backbone framework or inference process, making it applicable to various TSF models.

C. Representation Alignment with Supervised Representations

To enable the base predictor to learn richer temporal dependencies, ReGuider introduces an auxiliary representation supervision objective that aligns the encoder embedding of the base predictor with that of a pretrained foundation model.

Formally, given an input sequence $X \in \mathbb{R}^{C \times L}$, the base predictor $\mathcal{F}_\theta$ encodes it into a latent representation:

$$H_f = \mathcal{F}_\theta^{\text{enc}}(X), \quad (1)$$

while the foundation model $\mathcal{G}_\phi$ encodes the same sequence into:

$$H_g = \mathcal{G}_\phi^{\text{enc}}(X), \quad (2)$$

TABLE I
LONG TERM FORECASTING RESULTS WITH VARYING PREDICT LENGTHS $T \in \{96, 192, 336, 720\}$. THE HISTORICAL INPUT LENGTH L IS FIXED AT 96 FOR FAIR COMPARISON. THE BEST RESULTS ARE HIGHLIGHTED IN **BOLD**.

Models	iTransformer [11] + ReGuider				PatchTST [13] + ReGuider				DLinear [7] + ReGuider				TimeMixer [30] + ReGuider			
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.386	0.405	0.377 0.398	0.414	0.419	0.382 0.384	0.386	0.400	0.368 0.390	0.375	0.400	0.366 0.393			
	192	0.441	0.436	0.427 0.426	0.460	0.445	0.424 0.425	0.437	0.432	0.402 0.413	0.429	0.421	0.422 0.419			
	336	0.487	0.458	0.475 0.452	0.501	0.466	0.462 0.441	0.481	0.459	0.448 0.438	0.484	0.458	0.458 0.434			
	720	0.503	0.491	0.486 0.480	0.500	0.488	0.478 0.471	0.519	0.516	0.487 0.485	0.498	0.482	0.475 0.465			
ETTh2	96	0.297	0.349	0.289 0.343	0.302	0.348	0.293 0.338	0.333	0.387	0.320 0.361	0.289	0.341	0.382 0.334			
	192	0.380	0.400	0.373 0.392	0.388	0.400	0.374 0.387	0.477	0.476	0.406 0.424	0.372	0.392	0.358 0.384			
	336	0.428	0.432	0.415 0.427	0.426	0.433	0.412 0.421	0.594	0.541	0.453 0.455	0.386	0.414	0.379 0.410			
	720	0.427	0.445	0.420 0.441	0.431	0.446	0.418 0.429	0.831	0.657	0.596 0.541	0.412	0.434	0.406 0.427			
ETTm1	96	0.334	0.368	0.327 0.361	0.329	0.367	0.322 0.358	0.345	0.372	0.336 0.368	0.320	0.357	0.316 0.351			
	192	0.377	0.391	0.372 0.386	0.367	0.385	0.357 0.378	0.380	0.389	0.369 0.372	0.361	0.381	0.355 0.377			
	336	0.426	0.420	0.412 0.409	0.399	0.410	0.388 0.399	0.413	0.413	0.395 0.398	0.390	0.404	0.385 0.396			
	720	0.491	0.459	0.476 0.442	0.454	0.439	0.445 0.430	0.474	0.453	0.461 0.442	0.454	0.441	0.444 0.431			
ETTm2	96	0.180	0.264	0.175 0.258	0.175	0.259	0.168 0.248	0.193	0.292	0.173 0.269	0.175	0.258	0.170 0.252			
	192	0.250	0.309	0.242 0.300	0.241	0.302	0.234 0.287	0.284	0.362	0.263 0.348	0.237	0.299	0.233 0.296			
	336	0.311	0.348	0.303 0.339	0.305	0.343	0.301 0.335	0.369	0.427	0.344 0.401	0.298	0.340	0.291 0.330			
	720	0.412	0.407	0.401 0.396	0.402	0.400	0.386 0.392	0.554	0.522	0.472 0.493	0.391	0.396	0.387 0.390			
Weather	96	0.174	0.214	0.168 0.207	0.177	0.218	0.165 0.212	0.196	0.255	0.175 0.234	0.163	0.209	0.160 0.203			
	192	0.221	0.254	0.216 0.248	0.225	0.259	0.208 0.244	0.237	0.296	0.212 0.258	0.208	0.250	0.205 0.246			
	336	0.278	0.296	0.267 0.286	0.278	0.297	0.253 0.286	0.283	0.335	0.268 0.317	0.251	0.287	0.248 0.384			
	720	0.358	0.347	0.346 0.338	0.354	0.348	0.342 0.340	0.345	0.381	0.324 0.372	0.339	0.341	0.336 0.337			
ECL	96	0.148	0.240	0.143 0.236	0.181	0.270	0.163 0.256	0.197	0.282	0.166 0.269	0.153	0.247	0.152 0.245			
	192	0.162	0.253	0.158 0.248	0.188	0.274	0.169 0.263	0.196	0.285	0.179 0.277	0.166	0.256	0.164 0.256			
	336	0.178	0.269	0.173 0.265	0.204	0.293	0.195 0.286	0.209	0.301	0.196 0.289	0.185	0.277	0.182 0.272			
	720	0.225	0.317	0.209 0.298	0.246	0.324	0.227 0.309	0.245	0.333	0.214 0.318	0.225	0.310	0.222 0.308			
Traffic	96	0.395	0.268	0.379 0.262	0.462	0.295	0.405 0.272	0.650	0.396	0.521 0.293	0.462	0.285	0.397 0.270			
	192	0.417	0.276	0.402 0.268	0.466	0.296	0.420 0.278	0.598	0.370	0.546 0.308	0.473	0.296	0.422 0.281			
	336	0.433	0.283	0.431 0.276	0.482	0.304	0.434 0.292	0.605	0.373	0.552 0.329	0.498	0.296	0.441 0.290			
	720	0.467	0.302	0.454 0.295	0.514	0.322	0.474 0.310	0.645	0.394	0.568 0.343	0.506	0.313	0.483 0.297			

where $H_f, H_g \in \mathbb{R}^d$ denote the embeddings before the prediction head, and θ, ϕ are the parameters of the base predictor and foundation model, respectively.

We define the representation supervision loss as:

$$\mathcal{L}_{\text{TSRA}}(\theta, \phi) = \mathbb{E}_{X \sim \mathcal{D}} [\text{sim}(H_f, H_g)], \quad (3)$$

where $\text{sim}(\cdot, \cdot)$ is a similarity or distance function. Several options are possible:

Euclidean distance:

$$\text{sim}_{\ell_2}(H_f, H_g) = \|H_f - H_g\|_2^2. \quad (4)$$

Cosine similarity:

$$\text{sim}_{\text{cos}}(H_f, H_g) = 1 - \frac{H_f^\top H_g}{\|H_f\|_2 \|H_g\|_2}. \quad (5)$$

KL divergence:

$$\text{sim}_{\text{KL}}(H_f, H_g) = D_{\text{KL}}(\sigma(H_f) \parallel \sigma(H_g)), \quad (6)$$

where $\sigma(\cdot)$ denotes the softmax function.

The overall training objective combines the standard forecasting loss with the representation supervision loss:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{Pred}}(Y, \hat{Y}) + \lambda \mathcal{L}_{\text{TSRA}}(\theta, \phi), \quad (7)$$

where λ is a trade-off hyperparameter. This joint objective ensures that the predictor minimizes forecasting error while simultaneously learning embeddings guided by the pretrained foundation model.

A critical design choice in ReGuider is the asymmetric gradient flow. The parameters ϕ of the foundation model are frozen to preserve the universal temporal vocabulary:

$$\theta^*, \psi^* = \arg \min_{\theta, \psi} \mathcal{L}_{\text{Pred}}(Y, \hat{Y}) + \lambda \mathcal{L}_{\text{TSRA}}(\tilde{H}_f, \text{sp}(H_g)), \quad (8)$$

where $\text{sp}(\cdot)$ denotes the stop gradient operation. This ensures that the foundation model acts as a stationary semantic an-

TABLE II

RESULTS OF EUCLIDEAN DISTANCE (ED), KL DIVERGENCE (KLD), AND COSINE SIMILARITY (COS. SIM.) AS OPTIMIZATION OBJECTIVES FOR REPRESENTATION SUPERVISION.

Models		iTransformer [11]						PatchTST [13]					
		ED		KLD		Cos. Sim.		ED		KLD		Cos. Sim.	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	0.327	0.361	0.334	0.369	0.325	0.362	0.322	0.358	0.329	0.364	0.324	0.360
	192	0.372	0.386	0.379	0.392	0.373	0.388	0.357	0.378	0.363	0.385	0.359	0.378
	336	0.412	0.409	0.421	0.418	0.415	0.411	0.388	0.399	0.395	0.404	0.390	0.403
	720	0.476	0.442	0.482	0.451	0.479	0.445	0.445	0.430	0.452	0.435	0.447	0.431
Weather	96	0.168	0.207	0.173	0.213	0.169	0.207	0.165	0.212	0.171	0.214	0.166	0.210
	192	0.216	0.248	0.224	0.252	0.218	0.247	0.208	0.244	0.214	0.246	0.209	0.245
	336	0.267	0.286	0.276	0.292	0.270	0.287	0.253	0.286	0.259	0.288	0.254	0.285
	720	0.346	0.338	0.354	0.345	0.347	0.339	0.342	0.340	0.349	0.342	0.341	0.339
ECL	96	0.143	0.236	0.151	0.241	0.145	0.238	0.163	0.256	0.167	0.259	0.165	0.258
	192	0.158	0.248	0.166	0.253	0.160	0.247	0.169	0.263	0.175	0.266	0.170	0.262
	336	0.173	0.265	0.182	0.271	0.177	0.266	0.195	0.286	0.201	0.291	0.198	0.288
	720	0.209	0.298	0.217	0.305	0.210	0.297	0.227	0.309	0.234	0.314	0.228	0.310
Traffic	96	0.379	0.262	0.382	0.265	0.381	0.265	0.405	0.272	0.406	0.273	0.410	0.275
	192	0.402	0.268	0.404	0.269	0.406	0.271	0.420	0.278	0.421	0.277	0.424	0.280
	336	0.431	0.276	0.433	0.278	0.433	0.280	0.434	0.292	0.435	0.290	0.437	0.294
	720	0.454	0.295	0.456	0.296	0.460	0.299	0.474	0.310	0.475	0.309	0.478	0.312

chor, preventing the representation drift that often occurs in traditional co-training paradigms.

D. Discussions

Although ReGuider involves an architecture like teacher-student, it differs from traditional knowledge distillation (KD). Whereas conventional KD focuses on output alignment by mimicking the teacher’s final predictions or logits, ReGuider emphasizes representation alignment. For time series, we argue that the “teacher’s” value lies not in its specific forecast values, but in its universal temporal vocabulary. By aligning intermediate latent spaces, we avoid inheriting the teacher’s potential predictive biases and instead focus on enriching the student’s structural understanding.

IV. EXPERIMENTS

A. Setups

Dataset. We evaluate the proposed ReGuider model on 7 commonly used time series benchmark datasets: ETTh1, ETTh2, ETTm1, ETTm2, Weather, Electricity, and Traffic [24]. Specifically, the ETT series (ETTh1, ETTh2, ETTm1, ETTm2) records power load and oil temperature from electricity transformers at both hourly and 15-minute resolutions. The Weather dataset includes 21 meteorological indicators collected every 10 minutes, representing a typical low-dimensional physical sensing task. Electricity (ECL) tracks the hourly power consumption of 321 clients, serving as a mid-dimensional benchmark for demand forecasting. Finally, the Traffic dataset monitors hourly road occupancy rates from 862 sensors, providing a high-dimensional challenge for capturing complex spatial-temporal inter-dependencies. Consistent with

TABLE III

RESULTS OF TIME-MOE FOUNDATION MODEL—BASE, LARGE, AND ULTRA—AS REPRESENTATION GUIDERS.

Models	Metric	iTransformer [11]						PatchTST [13]					
		Base			Large			Base			Large		
		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTm1	96	0.325	0.361	0.327	0.361	0.334	0.368	0.324	0.357	0.322	0.358	0.336	0.370
	192	0.370	0.387	0.372	0.386	0.379	0.392	0.355	0.376	0.357	0.378	0.383	0.395
	336	0.414	0.412	0.412	0.409	0.420	0.416	0.386	0.400	0.388	0.399	0.423	0.419
	720	0.475	0.443	0.476	0.442	0.486	0.450	0.447	0.432	0.445	0.430	0.490	0.452
Weather	96	0.169	0.208	0.168	0.207	0.170	0.209	0.166	0.211	0.165	0.212	0.171	0.210
	192	0.217	0.247	0.216	0.248	0.219	0.249	0.208	0.243	0.208	0.244	0.210	0.246
	336	0.265	0.285	0.267	0.286	0.270	0.287	0.254	0.287	0.253	0.286	0.255	0.288
	720	0.345	0.340	0.346	0.338	0.348	0.339	0.343	0.341	0.342	0.340	0.344	0.342
ECL	96	0.148	0.240	0.143	0.236	0.152	0.243	0.167	0.259	0.163	0.256	0.160	0.255
	192	0.161	0.250	0.158	0.248	0.165	0.253	0.172	0.266	0.169	0.263	0.168	0.263
	336	0.178	0.270	0.173	0.265	0.181	0.272	0.196	0.287	0.195	0.286	0.192	0.284
	720	0.211	0.301	0.209	0.298	0.298	0.300	0.229	0.312	0.227	0.309	0.228	0.311
Traffic	96	0.391	0.270	0.379	0.262	0.376	0.263	0.412	0.276	0.405	0.272	0.405	0.270
	192	0.414	0.274	0.402	0.268	0.403	0.268	0.422	0.280	0.420	0.278	0.420	0.278
	336	0.439	0.280	0.431	0.276	0.433	0.275	0.436	0.291	0.434	0.292	0.432	0.289
	720	0.468	0.299	0.454	0.295	0.454	0.295	0.476	0.311	0.474	0.310	0.475	0.309

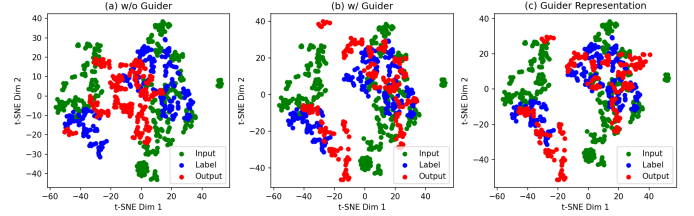


Fig. 2. Visualisation of the iTransformer’s and guider’s representations.

classic work [12], [36], we use Mean Squared Error (MSE) and Mean Absolute Error (MAE) as performance evaluation metrics.

Base Predictor and Base Representation Guider. For base predictor $\mathcal{F}(\cdot)$, it can be any mainstream deep learning-based time series forecasting model. We select four widely recognized models in LTSF literature: iTransformer [11], PatchTST [13], DLinear [7], and TimeMixer [30]. We compare their direct forecasting performance with their use as base predictors in our proposed ReGuider model. Furthermore, for foundation model $\mathcal{G}(\cdot)$ for Self-Supervised representation, we choose Time-MoE_{base} [19].

B. Main Results

As shown in Tab. I, ReGuider consistently improves forecasting performance when applied to various backbone predictors, including Transformer- and Linear-based architectures. Across four representative backbones, our method improves forecasting accuracy by over 5% on average, demonstrating its effectiveness in enriching temporal representations through representation-level supervision and confirming its generality as a seamless plug-in. This performance boost is particu-

larly pronounced in high-dimensional datasets such as Traffic, which contains 862 variables. In these complex scenarios, the alignment with a foundation model’s “universal temporal vocabulary” allows the base predictor to better capture intricate inter-variable couplings that are typically lost when training with error-only objectives, which tend to favor smoothed, uninformative averages.

Furthermore, ReGuider exhibits impressive stability as the forecasting horizon T increases from 96 to 720. In short-term forecasting ($T = 96$), the representation-level guidance assists the predictor in identifying sharp seasonality and abrupt regime shifts that point-wise losses often overlook. As the horizon extends to $T = 720$, where standard models typically suffer from a widening drifting latent states, ReGuider acts as a semantic anchor. By enforcing alignment with the foundation model’s stable embeddings (H_g), the student model sustains its predictive accuracy even at these challenging lengths, preventing the prediction from decaying toward a simple conditional mean.

C. Model Analysis

To further understand the behavior of ReGuider, we investigate the following research questions: **RQ1**: How should the distance between the base predictor’s embeddings and those of the foundation model be measured and optimized? **RQ2**: How do different foundation models perform when serving as representation guiders? **RQ3**: Does incorporating an additional guider significantly impact efficiency? **RQ4**: Can we observe clear richer representation under guidance?

RQ1. Distance metrics for supervision. We compare the use of Euclidean distance, KL divergence, and cosine similarity as optimization objectives for aligning the embeddings, and summarize the results in Tab. II. Of these three, Euclidean distance yields the best forecasting accuracy. This is because it directly measures point-wise closeness in the latent space, enforcing a tighter alignment between the two embedding distributions. In contrast, cosine similarity only constrains angular consistency without controlling magnitude, while KL divergence relies on distributional assumptions that may not hold in high-dimensional embedding spaces. Consequently, Euclidean distance provides the most stable and effective signal for representation supervision.

RQ2. Effect of different foundation models. We also evaluate three versions of the Time-MoE foundation model — base, large, and ultra — as representation guiders. As shown in Table 1, results indicate that different pretrained representations offer complementary strengths. For instance, on relatively small datasets such as ETT, the base version provides competitive guidance, demonstrating that lightweight models can effectively transfer temporal semantics. However, on larger, more complex datasets such as Traffic, the ultra variant with the highest parameter count achieves the best performance, highlighting the benefit of scaling foundation models to capture broader temporal patterns. These results suggest that the guider chosen should be adapted to the scale and complexity of the target dataset.

RQ3. Efficiency considerations. In terms of computational cost, ReGuider only requires the foundation model to be invoked during training in order to extract intermediate embeddings. While this introduces a marginal increase in training time, it does not affect inference since the foundation model is no longer needed once alignment is complete. Consequently, the method incurs negligible overhead at deployment, ensuring that the benefits of representation-level supervision are realised without any additional inference cost.

RQ4. Representation Visualization. To verify the effect of supervision, we feed a randomly selected window from the ETTm1 test set into both the vanilla iTransformer [11] encoder and the ReGuider-trained version of the same encoder. We then reduced the dimensions using t-SNE. As shown in Fig. 2, vanilla embeddings form a diffuse cloud with substantial overlap among trend classes, indicating weak temporal discrimination. After ReGuider supervision, the same encoder produces compact, well-separated clusters. This visual separation confirms that the guidance objective has transferred the foundation model’s rich temporal structure to the encoder, providing a more informative latent space.

V. CONCLUSION

In this study, we present ReGuider, a representation-level supervision method in the form of a plug-in for TSF. ReGuider enriches encoder embeddings by aligning them with representations extracted from pre-trained time series foundation models. This design allows forecasting architectures to capture richer temporal dependencies and semantic structures, delivering consistent performance enhancements across various backbones and datasets. Extensive experimentation confirms that ReGuider is effective and efficient, enhancing accuracy without incurring additional inference costs. Ultimately, We believe that this framework underscores the potential of foundation models as universal representation guides, opening new avenues for semantically aware temporal modeling.

VI. LLM USAGE DESCRIPTION

Large Language Models (LLMs) were used solely as additional tools for refining the language. The authors had full autonomy over all aspects of the research, including its conceptualisation, experimental execution and data interpretation. No AI was involved in the core scientific processes or the derivation of conclusions.

REFERENCES

- [1] Yifan Hu, Yuante Li, Peiyuan Liu, Yuxia Zhu, Naiqi Li, Tao Dai, Shu tao Xia, Dawei Cheng, and Changjun Jiang, “Fintsb: A comprehensive and practical benchmark for financial time series forecasting,” *arXiv preprint arXiv:2502.18834*, 2025.
- [2] Mustafa Gul and F Necati Catbas, “Statistical pattern recognition for structural health monitoring using time series modeling: Theory and experimental verifications,” *Mechanical Systems and Signal Processing*, vol. 23, pp. 2192–2204, 2009.
- [3] Zahra Karevan and Johan AK Suykens, “Transductive lstm for time-series prediction: An application to weather forecasting,” *Neural Networks*, vol. 125, pp. 1–9, 2020.

- [4] Yifan Hu, Guibin Zhang, Peiyuan Liu, Disen Lan, Naiqi Li, Dawei Cheng, Tao Dai, Shu-Tao Xia, and Shirui Pan, "Timefilter: Patch-specific spatial-temporal graph filtration for time series forecasting," in *Forty-second International Conference on Machine Learning*, 2025.
- [5] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, Xiaojun Chang, and Chengqi Zhang, "Connecting the dots: Multivariate time series forecasting with graph neural networks," in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 2020, pp. 753–763.
- [6] Abhimanyu Das, Weihao Kong, Andrew Leach, Shaan Mathur, Rajat Sen, and Rose Yu, "Long-term forecasting with TiDE: Time-series dense encoder," *arXiv preprint arXiv:2304.08424*, 2023.
- [7] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu, "Are transformers effective for time series forecasting?," in *Proceedings of the AAAI conference on artificial intelligence*, 2023, vol. 37, pp. 11121–11128.
- [8] Tao Dai, Beiliang Wu, Peiyuan Liu, Naiqi Li, Xue Yuerong, Shu-Tao Xia, and Zexuan Zhu, "Ddn: Dual-domain dynamic normalization for non-stationary time series forecasting," *Advances in Neural Information Processing Systems*, vol. 37, pp. 108490–108517, 2024.
- [9] Peiyuan Liu, Beiliang Wu, Yifan Hu, Naiqi Li, Tao Dai, Jigang Bao, and Shu-Tao Xia, "Timebridge: Non-stationarity matters for long-term time series forecasting," 2025.
- [10] Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin, "FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting," in *International Conference on Machine Learning*. PMLR, 2022, pp. 27268–27286.
- [11] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long, "iTransformer: Inverted transformers are effective for time series forecasting," *International Conference on Learning Representations*, 2024.
- [12] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long, "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting," *Advances in Neural Information Processing Systems*, vol. 34, pp. 22419–22430, 2021.
- [13] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam, "A time series is worth 64 words: Long-term forecasting with transformers," in *International Conference on Learning Representations*, 2023.
- [14] Debin Meng, Chen Jin, Zheng Gao, Yanran Li, Ioannis Patras, and Georgios Tzimiropoulos, "Training-free generation of diverse and high-fidelity images via prompt semantic space optimization," *arXiv preprint arXiv:2511.19811*, 2025.
- [15] Debin Meng, Christos Tzelepis, Ioannis Patras, and Georgios Tzimiropoulos, "Mm2latent: Text-to-facial image generation and editing in gans with multimodal assistance," in *European Conference on Computer Vision*. Springer, 2024, pp. 88–106.
- [16] Zheng Gao, Debin Meng, Yunqi Miao, Zhensong Zhang, Songcen Xu, Ioannis Patras, and Jifei Song, "Diffusion-based makeup transfer with facial region-aware makeup features," *arXiv preprint arXiv:2603.20012*, 2026.
- [17] Yifan Hu, Peiyuan Liu, Peng Zhu, Dawei Cheng, and Tao Dai, "Adaptive multi-scale decomposition framework for time series forecasting," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025, vol. 39, pp. 17359–17367.
- [18] Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Bernie Wang, "Chronos: Learning the language of time series," *Transactions on Machine Learning Research*, 2024, Expert Certification.
- [19] Xiaoming Shi, Shiyu Wang, Yuqi Nie, Dianqi Li, Zhou Ye, Qingsong Wen, and Ming Jin, "Time-moe: Billion-scale time series foundation models with mixture of experts," in *The Thirteenth International Conference on Learning Representations*, 2025.
- [20] Yong Liu, Haoran Zhang, Chenyu Li, Xiangdong Huang, Jianmin Wang, and Mingsheng Long, "Timer: Generative pre-trained transformers are large time series models," in *Forty-first International Conference on Machine Learning*, 2024.
- [21] Yu Tian, Zhongheng Yang, Chenshi Liu, Yiyun Su, Ziwei Hong, Zexi Gong, and Jingyuan Xu, "Centermamba-sam: Center-prioritized scanning and temporal prototypes for brain lesion segmentation," in *2025 IEEE 6th International Conference on Computer, Big Data, Artificial Intelligence (ICCBD+AI)*, 2025, pp. 1–5.
- [22] Yiyun Su, Huiying Zhu, Yu Tian, Changruo Zhao, Zujun Peng, Yuting Liu, Liang Fan, Baihua Li, and Luyan Zhang, "Agentic-sql taxonomy: A survey of autonomous and interactive text-to-sql with LLMs," *Authorea*, 2026.
- [23] Yifan Hu, Peiyuan Liu, Yuante Li, Dawei Cheng, Naiqi Li, Tao Dai, Jigang Bao, and Xia Shu-Tao, "Finmamba: Market-aware graph enhanced multi-level mamba for stock movement prediction," *arXiv preprint arXiv:2502.06707*, 2025.
- [24] Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long, "TimesNet: Temporal 2d-variation modeling for general time series analysis," in *International Conference on Learning Representations*, 2023.
- [25] Huiqiang Wang, Jian Peng, Feihu Huang, Jince Wang, Junhui Chen, and Yifei Xiao, "MICN: Multi-scale local and global context modeling for long-term series forecasting," in *The eleventh international conference on learning representations*, 2023.
- [26] Taesun Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo, "Reversible instance normalization for accurate time-series forecasting against distribution shift," in *International Conference on Learning Representations*, 2021.
- [27] Peiyuan Liu, Hang Guo, Tao Dai, Naiqi Li, Jigang Bao, Xudong Ren, Yong Jiang, and Shu-Tao Xia, "Calf: Aligning llms for time series forecasting via cross-modal fine-tuning," *arXiv preprint arXiv:2403.07300*, 2024.
- [28] Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie, "Representation alignment for generation: Training diffusion transformers is easier than you think," in *The Thirteenth International Conference on Learning Representations*, 2025.
- [29] Jingfeng Yao, Bin Yang, and Xinggang Wang, "Reconstruction vs. generation: Taming optimization dilemma in latent diffusion models," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 15703–15712.
- [30] Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Y Zhang, and Jun Zhou, "TimeMixer: Decomposable multiscale mixing for time series forecasting," *International Conference on Learning Representations*, 2024.
- [31] Yifan Hu, Jie Yang, Tian Zhou, Peiyuan Liu, Yujin Tang, Rong Jin, and Liang Sun, "Bridging past and future: Distribution-aware alignment for time series forecasting," *arXiv preprint arXiv:2509.14181*, 2025.
- [32] Gerald Woo, Chenghao Liu, Doyen Sahoo, Akshat Kumar, and Steven Hoi, "ETSformer: Exponential smoothing transformers for time-series forecasting," *arXiv preprint arXiv:2202.01381*, 2022.
- [33] Yushan Jiang, Wenchao Yu, Geon Lee, Dongjin Song, Kijung Shin, Wei Cheng, Yanchi Liu, and Haifeng Chen, "Timexl: Explainable multi-modal time series prediction with llm-in-the-loop," in *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.
- [34] Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou, "A decoder-only foundation model for time-series forecasting," in *Forty-first International Conference on Machine Learning*, 2024.
- [35] Abdul Fatir Ansari, Oleksandr Shchur, Jaris Küken, Andreas Auer, Boran Han, Pedro Mercado, Syama Sundar Rangapuram, Huibin Shen, Lorenzo Stella, Xiyuan Zhang, et al., "Chronos-2: From univariate to universal forecasting," *arXiv preprint arXiv:2510.15821*, 2025.
- [36] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *Proceedings of the AAAI conference on artificial intelligence*, 2021, vol. 35, pp. 11106–11115.